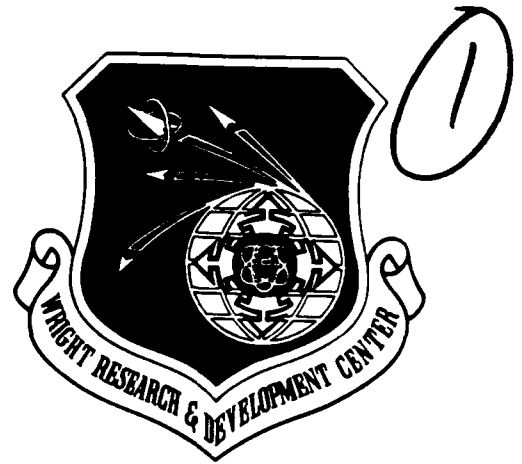


WRDC-TR-90-8007
Volume V
Part 2

AD-A252 447



INTEGRATED INFORMATION SUPPORT SYSTEM (IISS)
Volume V - Common Data Model Subsystem
Part 2 - CDMP Test Case Report

J. Althoff, M. Apicella, S. Singh

Control Data Corporation
Integration Technology Services
2970 Presidential Drive
Fairborn, OH 45324-6209

DTIC
ELECTE
JUN 03 1992
S B D

September 1990

Final Report for Period 1 April 1987 - 31 December 1990

Approved for Public Release; Distribution is Unlimited

92-14531



MANUFACTURING TECHNOLOGY DIRECTORATE
WRIGHT RESEARCH AND DEVELOPMENT CENTER
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6533

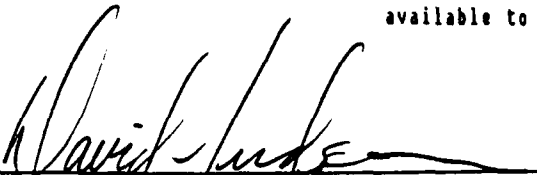
92 6 03 003

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever, regardless whether or not the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data. It should not, therefore, be construed or implied by any person, persons, or organization that the Government is licensing or conveying any rights or permission to manufacture, use, or market any patented invention that may in any way be related thereto.

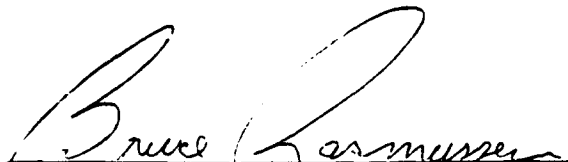
This technical report has been reviewed and is approved for publication.

This report is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations


DAVID L. JUDSON, Project Manager
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

FOR THE COMMANDER:


BRUCE A. RASMUSSEN, Chief
WRDC/MTI
Wright-Patterson AFB, OH 45433-6533

25 July 91
DATE

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify WRDC/MTI, Wright-Patterson Air Force Base, OH 45433-6533 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

MISSING PAGES WILL BE INSERTED AT AN LATER DATE
AS ERRATA(S)

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for Public Release; Distribution is Unlimited.		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) UTP620341000			5. MONITORING ORGANIZATION REPORT NUMBER(S) WRDC-TR-90-8007 Vol. V, Part 2		
6a. NAME OF PERFORMING ORGANIZATION Control Data Corporation; Integration Technology Services		6b. OFFICE SYMBOL (if applicable) WRDC/MTI		7a. NAME OF MONITORING ORGANIZATION WRDC/MTI	
6c. ADDRESS (City, State, and ZIP Code) 2970 Presidential Drive Fairborn, OH 45324-6209			7b. ADDRESS (City, State, and ZIP Code) WPAFB, OH 45433-6533		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Wright Research and Development Center, Air Force Systems Command, USAF		8b. OFFICE SYMBOL (if applicable) WRDC/MTI		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUM. F33600-87-C-0464	
8c. ADDRESS (City, State, and ZIP Code) Wright-Patterson AFB, Ohio 45433-6533			10. SOURCE OF FUNDING NOS.		
11. TITLE (Include Security Classification) See Block 19			PROGRAM ELEMENT NO. 78011F	PROJECT NO. 595600	TASK NO. F95600 WORK UNIT NO. 20950607
12. PERSONAL AUTHOR(S) Control Data Corporation: Althoff, J. L., Apicella, M. L., Singh, S.					
13a. TYPE OF REPORT Final Report		13b. TIME COVERED 4/1/87-12/31/90		14. DATE OF REPORT (Yr., Mo., Day) 1990 September 30	
15. PAGE COUNT 215					
16. SUPPLEMENTARY NOTATION WRDC/MTI Project Priority 6203					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify block no.)		
FIELD	GROUP	SUB GR.			
1308	0905				
19. ABSTRACT (Continue on reverse if necessary and identify block number) This document summarizes the set of test cases for the Common Data Model subsystem. Block 11 - INTEGRATED INFORMATION SUPPORT SYSTEM (IISS) Vol V - Common Data Model Subsystem Part 2 - CDMP Test Case Report					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED x SAME AS RPT. DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL David L. Judson			22b. TELEPHONE NO. (Include Area Code) (513) 255-7371		22c. OFFICE SYMBOL WRDC/MTI

FOREWORD

This technical report covers work performed under Air Force Contract F33600-87-C-0464, DAPro Project. This contract is sponsored by the Manufacturing Technology Directorate, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. It was administered under the technical direction of Mr. Bruce A. Rasmussen, Branch Chief, Integration Technology Division, Manufacturing Technology Directorate, through Mr. David L. Judson, Project Manager. The Prime Contractor was Integration Technology Services, Software Programs Division, of the Control Data Corporation, Dayton, Ohio, under the direction of Mr. W. A. Osborne. The DAPro Project Manager for Control Data Corporation was Mr. Jimmy P. Maxwell.

The DAPro project was created to continue the development, test, and demonstration of the Integrated Information Support System (IISS). The IISS technology work comprises enhancements to IISS software and the establishment and operation of IISS test bed hardware and communications for developers and users.

The following list names the Control Data Corporation subcontractors and their contributing activities:

<u>SUBCONTRACTOR</u>	<u>ROLE</u>
Control Data Corporation	Responsible for the overall Common Data Model design development and implementation, IISS integration and test, and technology transfer of IISS.
D. Appleton Company	Responsible for providing software information services for the Common Data Model and IDEF1X integration methodology.
ONTEK	Responsible for defining and testing a representative integrated system base in Artificial Intelligence techniques to establish fitness for use.
Simpact Corporation	Responsible for Communication development.
Structural Dynamics Research Corporation	Responsible for User Interfaces, Virtual Terminal Interface, and Network Transaction Manager design, development, implementation, and support.
Arizona State University	Responsible for test bed operations and support.

TABLE OF CONTENTS

		<u>Page</u>
SECTION	1.0 TEST CASES	1-1
	1.1 PRECOMPILER TEST CASES	1-1
	1.2 NDDL TEST CASES.	1-3
SECTION	2. TEST CASE REPORTS.	2-1
	2. 1 CDMQ1 - Precompilation/Generation of RP-Main	2-1
	2. 2 CDMQ1 - Runtime	2-3
	2. 3 CDMQ2 - Precompilation/Generation of RP-Main	2-7
	2. 4 CDMQ2 - Runtime	2-9
	2. 5 CDMQ3 - Precompilation/Generation of RP-Main	2-11
	2. 6 CDMQ3 - Runtime	2-13
	2. 7 CDMQ4 - Precompilation/Generation of RP-Main	2-14
	2. 8 CDMQ4 - Runtime	2-16
	2. 9 CDMQCRT - NDDL	2-20
	2.10 CDMQ5 - Precompilation/Generation of RP-Main	2-21
	2.11 CDMQ5 - Runtime	2-23
	2.12 CDMQ6 - Precompilation/Generation of RP-Main	2-27
	2.13 CDMQ6 - Runtime	2-29
	2.14 CDMQ7 - Precompilation/Generation of RP-Main	2-31
	2.15 CDMQ7 - Runtime	2-33
	2.16 CDMQ8 - Precompilation/Generation of RP-Main	2-34
	2.17 CDMQ8 - Runtime	2-36
	2.18 CDMQDRP - NDDL	2-40
	2.19 CDTVCR - NDDL	2-41
	2.20 CDTV - Precompilation/Generation of RP-Main	2-42
	2.21 CDTV - Runtime	2-45
	2.22 CDTVALT - NDDL	2-55
	2.23 CDTV - Precompilation/Generation of RP-Main	2-56
	2.24 CDTV - Runtime	2-59
	2.25 CDTVDRP - NDDL	2-67
	2.26 CDTICRT - NDDL	2-68
	2.27 CDTIC - Precompilation/Generation of RP-MAIN	2-69
	2.28 CDTIC - Runtime	2-72
	2.29 CDTIALT - NDDL	2-82
	2.30 CDTIF - Precompilation/Generation of RP-MAIN	2-83
	2.31 CDTIF - Runtime	2-86

2.32	CDTIDRP - NDDL.	2-94
2.33	UNIVCRT - NDDL.	2-95
2.34	CDUNI - Precompilation/Generation of RP-MAIN.	2-97
2.35	CDUNI - Runtime	2-101
2.36	UNIVDRP - NDDL.	2-105
2.37	NDDL01-Runtime.	2-106
2.38	NDDL02-Runtime.	2-108
2.39	NDDL03-Runtime.	2-109
2.40	NDDL04-Runtime.	2-110
2.41	NDDL05-Runtime.	2-111
2.42	NDDL06-Runtime.	2-112
2.43	NDDL07-Runtime.	2-113
2.44	NDDL08-Runtime.	2-114
2.45	NDDL09-Runtime.	2-115
2.46	NDDL10-Runtime.	2-116
2.47	NDDL11-Runtime.	2-117
2.48	NDDL12-Runtime.	2-120
2.49	NDDL13-Runtime.	2-121
2.50	NDDL14-Runtime.	2-122
2.51	NDDL15-Runtime.	2-123
2.52	NDDL16-Runtime.	2-124
2.53	NDDL17-Runtime.	2-125
2.54	NDDL18-Runtime.	2-126
2.55	NDDL19-Runtime.	2-128
2.56	NDDL20-Runtime.	2-129
2.57	NDDL21-Runtime.	2-130
2.58	NDDL22-Runtime.	2-131
2.59	NDDL23-Runtime.	2-132
2.60	NDDL24-Runtime.	2-133
2.61	NDDL25-Runtime.	2-134
2.62	NDDL26-Runtime.	2-235
2.63	NDDL27-Runtime.	2-236
2.64	NDDL28-Runtime.	2-237
2.65	NDDL29-Runtime.	2-238
2.66	NDDL30-Runtime.	2-239
2.67	NDDL31-Runtime.	2-240
2.68	NDDL32-Runtime.	2-241
2.69	NDDL33-Runtime.	2-242
2.70	NDDL34-Runtime.	2-243
2.71	NDDL35-Runtime.	2-244



Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

2.72	NDDL36-Runtime.	2-145 ✓
2.73	NDDL37-Runtime.	2-147
2.74	NDDL38-Runtime.	2-148
2.75	NDDL39-Runtime.	2-149
2.76	NDDL40-Runtime.	2-150
2.77	NDDL41-Runtime.	2-151
2.78	NDDL42-Runtime.	2-152
2.79	NDDL43-Runtime.	2-153
2.80	NDDL44-Runtime.	2-154
2.81	NDDL45-Runtime.	2-155
2.82	NDDL46-Runtime.	2-157
2.83	NDDL47-Runtime.	2-158
2.84	NDDL48-Runtime.	2-159
2.85	NDDL49-Runtime.	2-161
2.86	NDDL50-Runtime.	2-162
2.87	NDDL51-Runtime.	2-164
2.88	NDDL52-Runtime.	2-165
2.89	NDDL53-Runtime.	2-166
2.90	NDDL54-Runtime.	2-167
2.91	NDDL55-Runtime.	2-168
2.92	NDDL56-Runtime.	2-169
2.93	NDDL57-Runtime.	2-170
2.94	NDDL58-Runtime.	2-172
2.95	NDDL59-Runtime.	2-174
2.96	NDDL60-Runtime.	2-176
2.97	NDDL61-Runtime.	2-178
2.98	NDDL62-Runtime.	2-180
2.99	NDDL63-Runtime.	2-181
2.100	NDDL64-Runtime.	2-184
2.101	NDDL65-Runtime.	2-188
2.102	NDDL66-Runtime.	2-190

PREFACE

This preface contains 66 NDDL test cases and 36 Precompiler test cases that were used in testing the new functionality of the CDM Release 3.0 of IISS. Of the 36 precompiler test cases, the test cases that test usage of ORACLE and DB2 databases (CDTICRT, CDTIC, CDTIALT, CDTIF, and CDTIDRP) are not supported by the current hardware configuration. The IISS Test Bed Integration Procedure figure found in the Quality Assurance Plan was used as a guide for documenting all test cases. The following section contains a chart detailing the Precompiler and NDDL test cases.

SECTION 1
TEST CASES

1.1 Precompiler Test Cases

<u>TEST CASE</u>	<u>TYPE OF TEST CASE</u>	<u>TYPE OF DATABASE</u>	<u>PHASE OF THE CDM</u>
CDMQ1	COBOL	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQ2	COBOL	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQ3	COBOL	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQ4	COBOL	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQCRT	NDDL	CDM	NDDL
CDMQ5	FORTTRAN	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQ6	FORTTRAN	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQ7	FORTTRAN	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQ8	FORTTRAN	ORACLE	Precompilation/RP-MAIN Generation/Runtime
CDMQDRP	NDDL	CDM	NDDL
CDTVCRT	NDDL	CDM	NDDL
CDTVC	COBOL	ORACLE	Precompilation/RP_MAIN Generation/Runtime
CDTVALT	NDDL	CDM	NDDL
CDTVF	FORTTRAN	ORACLE	Precompilation/RP-MAIN Generation/Runtime
* CDTVDRP	NDDL	CDM	NDDL
* CDTICRT	NDDL	CDM	NDDL
* CDTIC	DB2	ORACLE	Precompilation/RP-MAIN Generation/Runtime
* CDTIALT	NDDL	CDM	NDDL

UTP620341000
30 September 1990

* CDTIF	FORTTRAN	ORACLE DB2	Precompilation/ RP-MAIN Generation/Runtime
* CDTIDRP	NDDL	CDM	NDDL
UNIVCRT	NDDL	CDM	NDDL
CDUNI	COBOL	ORACLE VAX-11	Precompilation/ RP-MAIN Generation/Runtime
UNIVDRP	NDDL	CDM	NDDL

* The hardware configuration will not support these test cases at this time.

1.2 NDDL Test Cases

<u>TEST CASE</u>	<u>NDDL COMMANDS</u>
NDDL01: CONCEPTUAL SCHEMA:	Create Model Create Attribute Create Entity Create Relation
NDDL02	Alter Model Alter Entity...Add Key Alter Relation
NDDL03	Describe Create Alias Alter Attribute ...Add Keyword Alter Attribute ...Add Domain
NDDL04	Alter Alias Drop Alias Alter Entity ...Drop Attribute Alter Attribute ...Drop Keyword Drop Keyword
NDDL05	Drop Relation Drop Attribute Drop Entity
NDDL06	Drop Model
NDDL07	Alter Attribute Ownership
NDDL08	Alter Entity ...Alter Key
NDDL09	Set Commit Halt with Rollback Commit Rollback
NDDL10 :DESCRIBE:	Create Description Type Create Model Create Attribute Describe
NDDL11	Describe Drop Model Drop Description Type Halt
NDDL12 :DOMAINS:	Create Domain

NDDL13		Alter Domain
NDDL14		Copy Domain
NDDL15		Drop Domain
NDDL16	:CONCEPTUAL SCHEMA:	Create Attribute Create Entity Create Relation Alter Entity
NDDL17	:CS/ES MAPPING:	Create View
NDDL18		Copy View
NDDL19		Drop View
	:INTERNAL SCHEMA:	
NDDL20	:ORACLE/DB2:	Define DBMS
NDDL21		Alter Host
NDDL22		Define Database Define Record
NDDL23		Alter Database
NDDL24	:CODASYL:	Define DBMS Define Host
NDDL25		Alter DBMS
NDDL26		Define Database Define Record Define Set
NDDL27		Alter Field
NDDL28	:TOTAL:	Define DBMS Define Host
NDDL29		Alter Host
NDDL30		Define Database Define Record
NDDL31		Alter Record Alter Field
NDDL32	:IMS:	Define DBMS Define PSB
NDDL33		Alter PSB
NDDL34		Define PCB Define Record

NDDL35		Alter Record
NDDL36	:CS/IS MAPPING:	Create Partition Create Union Define Module
NDDL37		Alter Partition Alter Union Alter Module
NDDL38		Create Map (Auc-Datafield) Define Algorithm
NDDL39		Alter Map
NDDL40		Create Map (Auc-Set) Create Map (Relation-Set)
NDDL41		Alter Map
NDDL42	:COPY CDM CONTENTS:	Copy Module
NDDL43		Copy DBMS
NDDL44		Copy Host
NDDL45		Copy Database
NDDL46		Copy Record
NDDL47		Copy Set
NDDL48		Copy Map
NDDL49	:DROP CS/IS MAPPINGS:	Drop Map
NDDL50		Drop Algorithm Drop Map Drop Module Drop Union Drop Partition
NDDL51	:DROP CONCEPTUAL SCHEMA:	Drop Entity Drop Attribute
NDDL52	:DROP INTERNAL SCHEMA:	Drop Field Drop Set Drop Record Drop Database
NDDL53		Drop PSB Drop DBMS Drop Host
NDDL54	:CONCEPTUAL SCHEMA:	Create Model(s)

	:MODELING COMMANDS:	Copy Description
NDDL55		Copy Attribute
NDDL56		Check Model
NDDL57		Compare Model
NDDL58		Copy Entity ...with Relation
NDDL59		Copy Entity ...with Structure
NDDL60		Copy Entity ...with Exceptions
NDDL61		Combine Entity
NDDL62		Copy Model
NDDL63		Copy Model ...with Exceptions
NDDL64		Merge Model
DDL65		
NDDL66		Drop Model

SECTION 2

TEST CASE REPORTS

2.1 CDMQ1 - Precompilation/Generation of RP-MAIN

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQ1 - Precompilation/Generation of RP-MAIN

Objective:

This test case consists of a COBOL Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals: 1

S/W Requirements:

ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
Conceptual/External Transform Generator
SQL Request Process Generator

Estimated Time for Test: 2-4 minutes

Special Resource Considerations:

The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ1.PRC.

Test Definition

Method of Performing Test:

With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs: \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

1. Enter Name Of Logical Unit Of Work: CDMQ1
2. Enter Name Of Host Where Application Will Run: VAX
3. Enter Desired Language of Generated Request Processors: COBOL
4. Enter Language of Source Program(s) (C/COBOL/FORTRAN): COBOL
5. Enter the type of Embedded Language Used (NDML/SQL): NDML
6. Enter Your CDM Username/Password: CDM/CDM
7. Enter Module Name Of Your Application: CDMQ1
8. Do You Want Obsolete Generated Code Deleted (Y/N)?: Y
9. Does The Application Access Any IBM Databases (Y/N)?: N
10. Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ1.PRC
11. Enter Name Of PRC (C/R To Stop, Include Extension):

BEGINNING PRECOMPILE
NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results:

The results of this test will consist of a modified Application Process, one Conceptual/External Transformer sub-program, two SQL Request Processor sub-programs, two Conceptual/Conceptual Transformer sub-programs and one SQL Request Processor main program.

Successful Completion Criteria for Test:

All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.2 CDMQ1 - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number : CDMQ1 - Runtime

Objective:

This test case will execute the precompiled COBOL Application Process containing an NDML Query Combination command. It will retrieve a list of attribute use class occurrences for attributes that are owned by an entity, but are not key in that owning entity for a given model. Two selects are required combined with a difference operator. The first select obtains all AUC occurrences for a given model, going through attribute class to limit to the owner occurrences. The second select obtains all key class members for a given model.

Resource Requirements

Number of terminals: 2

S/W Requirements:

ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
Application processor executable (CDMQ1)

Estimated Time for Test: 3 minutes

Special Resource Considerations:

Test CDMQ1 - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of Performing Test

```
$ RUN CDMQ1ZZZ
```

```
*****  
ENTER MODEL NAME  
*****  
INTEGRATED_MODEL
```

Expected Test Results: Test results will be shown below.

Successful Completion Criteria for Test:

EC-NO	TAG-NO
1	128
2	123
2	129
3	2708
4	7
4	8
4	9
5	166
7	206
7	207
7	2707
9	219
10	12
10	13
10	14
11	1016
12	150
13	19
13	182
16	24
17	26
17	27
17	28
17	29
17	139
17	140
18	30
19	271
20	31
22	33
22	246
22	248
23	215
24	35
24	36
24	37
24	38
24	40
25	44
26	46
26	47
26	153
28	50
28	205
29	52
29	187
29	1014
29	1015
31	58
31	59
31	60
31	61
32	62
32	191
33	121

34	66
34	67
34	149
36	224
37	69
39	72
39	73
39	74
39	122
39	124
41	192
42	78
42	204
43	216
46	80
46	81
47	83
47	242
48	151
49	85
49	197
51	134
52	88
53	90
54	162
60	96
60	97
60	98
60	99
64	170
65	222
77	273
78	107
78	167
78	308
79	117
79	141
79	276
80	263
81	102
81	103
81	104
81	105
81	106
81	309
81	312
82	274
82	275
82	295
1003	1011
1004	1003
1004	1004
1004	1005
1004	1006
1014	1050
1014	1051
1016	1057
1017	1062
1017	1063

UTP620341000
30 September 1990

1017	1064
1017	1065
1018	1069
1018	1070
1019	1075
1019	1076
1019	1077
1122	2592
1122	2593
1122	2594
1122	2595
1123	2596
1123	2597
1123	2598
1123	2599
1124	2600
1124	2601
1124	2602
1124	2603
1124	2604
1156	2689
1157	2692
1157	2695
1158	2695
1158	2698
1159	2699
1160	2709
1160	2714
1160	2715
1161	2716
1161	2717
1161	2718
1161	2719
1161	2720
1161	2721
1161	2722
1161	2723

2.3 CDMQ2 - Precompilation/Generation of RP-MAIN

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQ2 - Precompilation/Generation of RP-MAIN

Objective:

This test case consists of a COBOL Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals : 1

S/W Requirements:

ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
Conceptual/External Transform Generator
SQL Request Process Generator

Estimated Time for Test: 2-4 minutes

Special Resource Considerations:

The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ2.PRC.

Test Definition

Method of Performing Test:

With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs: \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDMQ2
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: COBOL
Enter language of Source Program(s) (C/COBOL/FORTRAN): COBOL
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDMQ2
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ2.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

BEGINNING PRECOMPILE
NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results:

The results of this test will consist of a modified Application Process, two Conceptual/External Transformer sub-programs, three SQL Request Processor sub-programs, two Conceptual/Conceptual Transformer sub-programs and one SQL Request Processor main program.

Successful Completion Criteria for Test:

All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.4 CDMQ2 - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQ2 - Runtime

Objective:

This test case will execute the precompiled COBOL Application Process containing an NDML Query Combination command. It will determine the unique list of NDML modules associated with a given view name. It will use a select distinct on the combination of two searches unioned together, one for the VIEW_USAGE cross reference and the other for the DATA_ITEM_USAGE.

Resource Requirements

Number of terminals: 2

S/W Requirements:

ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
Application processor executable (CDMQ2)

Estimated Time for Test: 3 minutes

Special Resource Considerations:

Test CDMQ2 - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of Performing Test

```
$ RUN CDMQ2ZZZ  
  
*****  
ENTER VIEW NAME  
*****  
DOMAIN_RANGE
```

Expected Test Results: Test results will be shown below.

Successful Completion Criteria for Test:

	MOD-NAME: ALLRNG	LUW-NAME: NDDL
	PRECOMP-DATE: 26-AUG-87	STATUS-IND: N

	MOD-NAME:	
CHKRNG	LUW-NAME: NDDL	
	PRECOMP-DATE: 17-SEP-87	STATUS-IND: N

	MOD-NAME:	
DRPRNG	LUW-NAME: NDDL	
	PRECOMP-DATE: 27-AUG-87	STATUS-IND: N

	MOD-NAME:	
DRPRNGA	LUW-NAME: NDDL	
	PRECOMP-DATE: 27-AUG-87	STATUS-IND: N

	MOD-NAME:	
INSRNG	LUW-NAME: NDDL	
	PRECOMP-DATE: 27-AUG-87	STATUS-IND: N

	MOD-NAME:	
RETRNGA	LUW-NAME: NDDL	
	PRECOMP-DATE: 27-AUG-87	STATUS-IND: N

	MOD-NAME:	
VERRNG	LUW-NAME: NDDL	
	PRECOMP-DATE: 28-AUG-87	STATUS-IND: N

2.5 CDMQ3 - Precompilation/Generation of RP-MAIN

Subsystem: CDM

Release: 3.0

Test Name and Number:

CDMQ3 - Precompilation/Generation of RP-MAIN

Objective:

This test case consists of a COBOL Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals: 1

S/W Requirements:

ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
Conceptual/External Transform Generator
SQL Request Process Generator

Estimated Time for Test: 2-4 minutes

Special Resource Considerations:

The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ3.PRC.

Test Definition

Method of Performing Test:

With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

2.6 CDMQ3 - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQ3 - Runtime

Objective:

This test case will execute the precompiled COBOL Application Process containing an NDML Query Combination command. It is a two-level query combination which will determine those keywords that are used for both entities and attributes, regardless of model or for relations anywhere. It will union the result of an intersection of two selects with a third select.

Resource Requirements

Number of terminals: 2

S/W Requirements:

ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
Application processor executable (CDMQ3)

Estimated Time for Test: 3 minutes

Special Resource Considerations:

Test CDMQ3 - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of Performing Test

\$ RUN CDMQ3ZZZ

Expected Test Results: Test results will be shown below.

Successful Completion Criteria for Test: KEYWORD CDM-1

2.7 CDMQ4 - Precompilation/Generation of RP-MAIN

Subsystem: CDM

Release: 3.0

Test Name and Number:

CDMQ4 - Precompilation/Generation of RP-MAIN

Objective:

This test case consists of a COBOL Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals: 1

S/W Requirements:

ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
Conceptual/External Transform Generator
SQL Request Process Generator

Estimated Time for Test: 2-4 minutes

Special Resource Considerations:

The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ4.PRC.

Test Definition

Method of Performing Test:

With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs: \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDMQ4
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: COBOL
Enter Language of Source Program(s) (C/COBOL/FORTRAN): COBOL
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDMQ4
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ4.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

BEGINNING PRECOMPILE
NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results:

The results of this test will consist of a modified Application Process, two Conceptual/External Transformer sub-programs, two SQL Request Processor sub-programs, and a one SQL Request Processor main program.

Successful Completion Criteria for Test:

All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.8 CDMQ4 - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQ4 - Runtime

Objective:

This test case will execute the precompiled COBOL Application Process containing an NDML Query Combination command. It will retrieve a key-class number and use that data as input to retrieve all the tag numbers for that key class number in another select. It then retrieves another key-class number and repeats the process. This test case uses an inner select within the curly brackets.

Resource Requirements

Number of terminals: 2

S/W Requirements:

ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
Application processor executable (CDMQ4)

Estimated Time for Test: 3 minutes

Special Resource Considerations:

Test CDMQ4 - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of Performing Test

\$ RUN CDMQ4ZZZ

Expected Test Results: Test results will be shown below.

Successful Completion Criteria for Test:

TAG NUMBER	KEY CLASS NUMBER
93	11
94	12
16	1
15	2
21	3
22	4
23	5
32	6
34	7
68	8
70	9

71	10
4	24
6	25
95	13
17	14
18	15
112	15
45	17
116	17
1	18
114	19
118	19
2	20
65	21
75	22
76	23
57	38
188	38
189	38
171	39
172	39
200	39
25	26
146	27
100	28
101	28
143	28
11	29
53	30
54	31
158	31
77	32
159	32
154	34
185	34
64	36
190	36
56	37
188	37
189	37
125	52
145	52
87	53
147	53
55	40
198	42
208	42
209	42
41	44
42	44
43	44
120	44
109	45
130	45
110	46
111	46
131	46
3	48
135	48

UTP620341000
30 September 1990

5	49
136	49
137	49
10	50
138	50
89	51
144	51
79	65
174	65
175	65
51	66
178	66
179	66
126	54
148	54
160	55
127	57
2706	57
165	58
173	59
82	60
176	60
177	60
20	61
183	61
184	61
186	62
108	63
142	63
194	64
193	64
195	64
196	64
272	94
298	95
168	67
169	67
213	67
214	67
217	68
218	69
220	70
221	71
223	72
225	73
226	73
227	73
84	77
180	77
181	77
243	78
244	78
245	78
2705	78
210	79
211	79
212	79
247	79
1008	1001

1009	1001
1010	1002
1012	1002
299	96
302	96
63	97
306	98
305	99
307	99
310	100
311	101
296	102
301	102
297	103
300	103
303	104
304	104
1002	1000
1007	1000
2691	1138
2694	1138
2696	1139
2697	1139
1013	1003
1052	1016
1053	1017
1058	1017
1056	1018
1059	1018
1060	1018
1061	1018
1066	1019
1067	1019
1068	1019
1071	1020
1072	1020
1073	1020
1074	1020
1078	1021
1079	1021
1080	1021
2690	1136
2693	1137
2704	1141
2710	1142
2711	1142
2712	1142
2713	1143

2.9 CDMQCRT - NDDL

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQCRT - NDDL

Objective:

This test case is to be run before CDMQ5 Precompilation. It creates new data types needed to run the FORTRAN counterparts of CDMQ1 through CDMQ4.

Resource Requirements

Number of terminals: 1

S/W Requirements:

ORACLE CDM DATABASE
CDMP Distributed Request Supervisor
NDDL Command Processors

Estimated Time for Test: 5-10 minutes

Special Resource Considerations:

This test case requires the following data file containing the NDDL commands: CDMQCRT.DAT

Test Definition

Method of Performing Test:

With ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ NDDL CDMQCRT

Expected Test Results:

The CDM is populated with information necessary to perform the NDML precompiler test cases.

Successful Completion Criteria for Test:

The execution status will be reported as successful for each NDDL command tested in CDMQCRT.OUT

2.10 CDMQ5 - Precompilation/Generation of RP-MAIN

Subsystem: CDM

Release: 3.0

Test Name and Number: CDMQ5 - Precompilation/Generation of RP-MAIN

Objective:

This test case consists of a FORTRAN Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals: 1

S/W Requirements:

ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
Conceptual/External Transform Generator
SQL Request Process Generator

Estimated Time for Test: 2-4 minutes

Special Resource Considerations:

The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ5.PRC.

Test Definition

Method of Performing Test:

With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs: \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDMQ5
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: FORTRAN
Enter Language of Source Program(s) (C/COBOL/FORTRAN): FORTRAN
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDMQ5
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ5.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results:

The results of this test will consist of a modified Application Process, one Conceptual/External Transformer sub-program, two SQL Request Processor sub-programs, two Conceptual/Conceptual Transformer sub-programs and one SQL Request Processor main program.

Successful Completion Criteria for Test:

All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.11 CDMQ5 - Runtime

Subsystem : CDM

Release: 3.0

Test Name and Number : CDMQ5 - Runtime

Objective: This test case will execute the precompiled FORTRAN Application Process containing an NDML Query Combination command. It will retrieve a list of attribute use class occurrences for attributes that are owned by an entity, but are not key in that owning entity for a given model. Two selects are required combined with a difference operator. The first select obtains all AUC occurrences for a given model, going through attribute class to limit to the owner occurrences. The second select obtains all key class members for a given model.

Resource Requirements

Number of terminals : 2

S/W Requirements : ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
Application processor executable (CDMQ5)

Estimated Time for Test : 3 minutes

Special Resource Considerations: Test CDMQ5 - Precompilation/
Generation of RP-MAIN must have successfully completed.

Test Definition

Method of Performing Test

\$ RUN CDMQ5ZZZ

ENTER MODEL NAME

INTEGRATED_MODEL

Expected Test Results: Test results will be shown below.

Successful Completion Criteria for Test

EC-NO	TAG-NO
1	128
2	123
2	129
3	2708
4	7
4	8
4	9
5	166
7	206
7	207
7	2707

UTP620341000
30 September 1990

9	219
10	12
10	13
10	14
11	1016
12	150
13	19
13	182
16	24
17	26
17	27
17	28
17	29
17	139
17	140
18	30
19	271
20	31
22	33
22	246
22	248
23	215
24	35
24	36
24	37
24	38
24	40
25	44
26	46
26	47
26	153
28	50
28	205
29	52
29	187
29	1014
29	1015
31	58
31	59
31	60
31	61
32	62
32	191
33	121
34	66
34	67
34	149
36	224
37	69
39	72
39	73
39	74
39	122
39	124
41	192
42	78
42	204
43	216
46	80

46	81
47	83
47	242
48	151
49	85
49	197
51	134
52	88
53	90
54	162
60	96
60	97
60	98
60	99
64	170
65	222
77	273
78	107
78	167
78	308
79	117
79	141
79	276
80	263
81	102
81	103
81	104
81	105
81	106
81	302
81	312
82	274
82	275
1003	1011
1004	1003
1004	1004
1004	1005
1004	1006
1004	1050
1014	1051
1016	1057
1017	1062
1017	1063
1017	1064
1017	1065
1018	1069
1018	1070
1019	1075
1019	1076
1019	1077
1122	2592
1122	2593
1122	2594
1123	2595
1123	2596
1123	2597
1123	2598
1123	2599
1124	2600

UTP62C341000
30 September 1990

1124	2601
1124	2602
1124	2603
1124	2604
1156	2689
1157	2692
1157	2695
1158	2698
1158	2699
1159	2709
1160	2714
1160	2715
1160	2716
1161	2717
1161	2718
1161	2719
1161	2720
1161	2721
1161	2722
1161	2723
1196	2749
1196	2750
1196	2751
1196	2752
1196	2753
1196	2754
1196	2756
1198	2758
1198	2759
1199	2760
1199	2762

2.12 CDMQ6 - Precompilation/Generation of RP-MAIN

Subsystem : CDM Release: 3.0

Test Name and Number : CDMQ6 - Precompilation/Generation
of RP-MAIN

Objective: This test case consists of a FORTRAN Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: CDM File/Module Processing
Capabilities
IISS Precompiler
Conceptual/External Transform
Generator
SQL Request Process Generator

Estimated Time for Test : 2-4 minutes

Special Resource

Considerations : The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ6.PRC.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDMQ6
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: FORTRAN
Enter Language of Source Program(s) (C/COBOL/FORTRAN): FORTRAN
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDMQ6
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ6.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results:

The results of this test will consist of a modified Application Process, one Conceptual/External Transformer sub-program, two SQL Request Processor sub-programs, two Conceptual/Conceptual Transformer sub-programs and one SQL Request Processor main program.

Successful Completion Criteria for Test

All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.13 CDMQ6 - Runtime

Subsystem : CDM

Release: 3.0

Test Name and Number : CDMQ6 - Runtime

Objective: This test case will execute the precompiled FORTRAN Application Process containing an NDML Query Combination command. It will determine the unique list of NDML modules associated with a given view name. It will use a select distinct on the combination of two searches unioned together, one for the VIEW_USAGE cross reference and the other for the DATA_ITEM_USAGE.

Resource Requirements

Number of terminals : 2

S/W Requirements : ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
Application processor executable (CDMQ6)

Estimated Time for Test : 3 minutes

Special Resource

Considerations : Test CDMQ6 - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of
Performing Test

\$ RUN CDMQ6ZZZ

ENTER VIEW NAME

DOMAIN_RANGE

Expected Test Results : Test results will be shown below.

UTP620341000
30 September 1990

Successful Completion
Criteria for Test

: MOD-NAME: ALLRNG LUW-NAME: NDDL
PRECOMP-DATE: 26-AUG-87 STATUS-IND: N

MOD-NAME:
CHKRNG LUW-NAME: NDDL
PRECOMP-DATE: 17-SEP-87 STATUS-IND: N

MOD-NAME:
DRPRNG LUW-NAME: NDDL
PRECOMP-DATE: 27-AUG-87 STATUS-IND: N

MOD-NAME:
DRPRNGA LUW-NAME: NDDL
PRECOMP-DATE: 27-AUG-87 STATUS-IND: N

MOD-NAME:
INSRNG LUW-NAME: NDDL
PRECOMP-DATE: 27-AUG-87 STATUS-IND: N

MOD-NAME:
RETRNGA LUW-NAME: NDDL
PRECOMP-DATE: 27-AUG-87 STATUS-IND: N

MOD-NAME:
VERRNG LUW-NAME: NDDL
PRECOMP-DATE: 28-AUG-87 STATUS-IND: N

2.14 CDMQ7 - Precompilation/Generation of RP-MAIN

Subsystem : CDM Release: 3.0

Test Name and Number : CDMQ7 - Precompilation/Generation
of RP-MAIN

Objective: This test case consists of a FORTRAN Application Process containing a query-combination NDML request. The query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
Conceptual/External Transform
Generator
SQL Request Process Generator

Estimated Time for Test : 2-4 minutes

Special Resource Considerations: The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ7.PRC.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDMQ7
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processor: FORTRAN
Enter Language of Source Program(s) (C/COBOL/FORTRAN): FORTRAN
Enter the type of Embedded Language Used: NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDMQ7
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ7.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test will consist of a modified Application Process, one Conceptual/External Transformer sub-program, three SQL Request Processor sub-programs, three Conceptual/Conceptual Transformer sub-programs and one SQL Request Processor main program.

Successful Completion
Criteria for Test : All of the above generated programs and subprograms will compile free of warnings and fatal errors.

2.15 CDMQ7 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : CDMQ7 - Runtime

Objective: This test case will execute the precompiled FORTRAN Application Process containing an NDML Query Combination command. It is a two-level query combination which will determine those keywords that are used for both entities and attributes, regardless of model or for relations anywhere. It will union the result of an intersection of two selects with a third select.

Resource Requirements

Number of terminals : 2

S/W Requirements : ORACLE CDM DATABASE
NTM: Message and queue server capabilities
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
Application processor executable
(CDMQ7)

Estimated Time for Test : 3 minutes

Special Resource

Considerations : Test CDMQ7 - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of
Performing Test

\$ RUN CDMQ7ZZZ

Expected Test Results : Test results will be shown below.

Successful Completion
Criteria for Test

: KEYWORD = CDM-1

2.16 CDMQ8 - Precompilation/Generation of RP-MAIN

Subsystem : CDM Release: 3.0

Test Name and Number : CDMQ8 - Precompilation/Generation
of RP-MAIN

Objective: This test case consists of a FORTRAN Application Process containing a query-combination NDML request the query will request data from the ORACLE/CDM database. This test case will precompile, compile, generate request processor main programs and link the application.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: CDM File/Module Processing

Capabilities
IISS Precompiler
Conceptual/External Transform

Generator
SQL Request Process Generator

Estimated Time for Test : 2-4 minutes

Special Resource

Considerations : The test case requires an Application Process with an embedded NDML request. The source file for this Application Process is CDMQ8.PRC.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDMQ8
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: FORTRAN
Enter Language of Source Program(s) (C/ COBOL/FORTRAN): FORTRAN
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDMQ8
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDMQ8.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test will consist of a modified Application Process, two Conceptual/External Transformer sub-programs, two SQL Request Processor sub-programs, and one SQL Request Processor main program.

Successful Completion
Criteria for Test : All of the above generated programs and subprograms will compile free of warning and fatal errors.

112	15
45	17
116	17
1	18
114	19
118	19
2	20
65	21
75	22
76	23
57	38
188	38
189	38
171	39
172	39
200	39
25	26
146	27
100	28
101	28
143	28
11	29
53	30
54	31
158	31
77	32
159	32
154	34
185	34
64	36
190	36
56	37
188	37
189	37
125	52
145	52
87	53
147	53
55	40
198	42
208	42
209	42
41	44
42	44
43	44
120	44
109	45
130	45
110	46
111	46
131	46
3	48
135	48
5	49
136	49
137	49
10	50
138	50
89	51

UTP620341000
30 September 1990

144	51
79	65
174	65
175	65
51	66
178	66
179	66
126	54
148	54
160	55
127	57
2706	57
165	58
173	59
82	60
176	60
177	60
20	61
183	61
184	61
186	62
108	63
142	63
194	63
193	64
195	64
196	64
272	94
298	95
168	67
169	67
213	67
214	67
217	68
218	69
220	70
221	71
223	72
225	73
226	73
227	73
84	77
180	77
181	77
243	78
244	78
245	78
2705	78
210	79
211	79
212	79
247	79
1008	1001
1009	1001
1010	1002
1012	1002
299	96
302	96
63	97

306	98
305	99
307	99
310	100
311	101
296	102
301	102
297	103
300	103
303	104
304	104
1002	1000
1007	1000
2691	1138
2694	1138
2696	1139
2697	1139
1013	1003
1052	1016
1053	1017
1058	1017
1056	1018
1059	1018
1060	1018
1061	1018
1066	1019
1067	1019
1068	1019
1071	1020
1072	1020
1073	1020
1074	1020
1078	1021
1079	1021
1080	1021
2690	1136
2693	1137
2704	1141
2710	1142
2711	1142
2712	1142
2713	1143
2755	1152
2757	1153
2761	1154
2763	1154
2764	1155
2765	1155
2766	1155

2.18 CDMQDRP - NDDL

Subsystem : CDM Release: 3.0

Test Name and Number : CDMQDRP - NDDL

Objective: This test case is to be run after CDMQ8 - Runtime. It drops the data types created in CDMQCRT for the four FORTRAN combination query test cases.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP Distributed Request Supervisor
NDDL Command Processors

Estimated Time for Test : 5-10 minutes

Special Resource

Considerations : This test case requires the following data file containing the NDDL commands:

CDMQDRP.DAT

Test Definition

Method of
Performing Test : With ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ NDDL CDMQDRP

Expected Test Results : The selected data types are
dropped from the CDM.

Successful Completion

Criteria for Test : The execution status will be
reported as successful for each NDDL command tested in CDMQDRP.OUT

2.19 CDTVCRT - NDDL

Subsystem : CDM Release: 3.0

Test Name and Number : CDTVCRT - NDDL

Objective: This test case executes NDDL Conceptual, External, Internal, Map, and Complex Map commands to populate the CDM with meta data required for the CDTV (Vax COBOL) series of precompile and runtime test cases.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE

CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL Command Processors

Estimated Time for Test : 35-40 minutes

Special Resource
Considerations : This test case requires the following data file which contains the NDDL commands:

CDTVCRT.DAT

Test Definition

Method of
Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @NDDL CDTVCRT

Expected Test Results : The CDM is populated with the information necessary to perform the CDTV precompiler test cases.

Successful Completion Criteria for Test : The execution status will be reported as successful for each NDDL command listed in: CDTVCRT.OUT

2.20 CDTVC - Precompilation/Generation of RP-Main

Subsystem : CDM Release: 3.0

Test Name and Number : CDTVC - Precompilation/Generation
of RP-Main

Objective: This test case consists of eleven Application Processes containing NDML requests as a single logical unit of work. It will test the retrieval, insert, modify and delete capabilities of the IISS precompiler. The test case will precompile, compile, generate request processor main programs and link the applications.

CDTVC01: CDTVC01 tests the insert action of the precompiler and the generation of calls to a verification module, a conceptual-to-internal complex mapping, and an external-to-conceptual complex mapping. It also tests the capability of several NDML requests in one application process and the capability to insert values from a file as well as values included in the application program.

CDTVC02: CDTVC02 tests the use of the statistical functions AVG, MIN, MAX, SUM, and COUNT.

CDTVC03: CDTVC03 tests the use of SELECT INTO STRUCTURE and the use of ALL. It also generates a call to an internal-to-conceptual complex mapping.

CDTVC04: CDTVC04 tests the use of SELECT INTO FILE and the use of ORDER BY DESCENDING. It also generates a call to a conceptual-to-external complex mapping.

CDTVC05: CDTVC05 tests the use of parenthetical logic separated with AND and OR operators. It generates calls to conceptual-to-external and internal-to-conceptual complex mappings. It also tests the use of the BETWEEN operator. This NDML retrieval command selects data distributed across two ORACLE databases.

CDTVC06: CDTVC06 tests the use of the Outerjoin (U=) operator with a single NDML query across two ORACLE databases. It also generates a call to a conceptual-to-external complex mapping.

CDTVC07: CDTVC07 tests the use of the query combination command with the UNION operator across two ORACLE databases, also using the BETWEEN operator. It generates calls to a conceptual-to-external and an internal-to-conceptual complex mapping.

CDTVC08: CDTVC08 tests the use of the XOR operator in a single NDML query across two ORACLE databases. It also generates a conceptual-to-external and an internal-to-conceptual complex mapping.

CDTVC09: CDTVC09 tests the use of the NDML modify command.

CDTVC10: CDTVC10 uses the NDML query capabilities to retrieve the information that was updated in the previous test - CDTVC09.

CDTVC11: CDTVCl1 tests the use of several NDML DELETE commands within one application process. One delete will use the BETWEEN operator. It also generates a call to a conceptual-to-internal complex mapping.

Resource Requirements

Number of terminals : 1
S/W Requirements : ORACLE CDM DATABASE
CDMP: CDM File/Module Processing Capabilities
IISS Precompiler
SQL Request Processor Generators

Estimated Time for Test : 60 minutes

Special Resource Considerations : This test case requires that the verification module generated in CDTVCRt - NDDL, the two complex mapping algorithm programs and the application driver program be compiled:

\$@ COBGLIB MDRLX.COB
\$@ COBGLIB MDRLX2.COB
\$@ COBGLIB CDTVCDR.COB

This test case requires eleven (11) Application Processes with NDML requests. The source files for these Application Processes are: CDTVCO1.PRC, CDTVCO2.PRC, CDTVCO3.PRC, CDTVCO4.PRC, CDTVCO5.PRC, CDTVCO6.PRC, CDTVCO7.PRC, CDTVCO8.PRC, CDTVCO9.PRC, CDTVCO10.PRC, and CDTVCl1.PRC. These Application Processes must be precompiled as a single logical unit of work.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDTVc
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: COBOL
Enter Language of Source Program(s) (C/COBOL/FORTRAN): COBOL
Enter the type of Embedded Language Used (NDML/SQL): NDML

Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDTVCDR
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? : N
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV01.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV02.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV03.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV04.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV05.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV06.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV07.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV08.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV09.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV10.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV11.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test will be displayed on the user terminal as indicated above. It will consist of eleven (11) modified Application Processes, forty (40) SQL Request Processor subprograms, sixteen (16) Conceptual/Conceptual Transformer subprograms, eight (8) Conceptual/External Transformer subprograms, and two Request Processor main programs.

Successful Completion Criteria for Test : All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.21 CDTVC - Runtime

Subsystem : CDM

Release: 3.0

Test Name and Number : CDTVC - Runtime

Objective: This test case will execute the precompiled COBOL Application Processes containing NDML select, insert, modify and delete requests against the ORACLE TEAM and ORACLE PLAYER databases. Eleven application processes have been grouped together as a single logical unit of work - one inserts into both databases, eight retrieve data from one or both databases, one modifies the PLAYER database, and one finally deletes from both databases. A driver program is also used.

Resource Requirements

Number of terminals : 2

S/W Requirements : ORACLE TEAM and PLAYER databases. These database must have been created using the procedures provided in the "CDM Subsystem Database Build Instructions" Document.

NTM: Message and queue server

CDMP: Distributed Request Supervisor CDM File/Module Processing Capabilities Application driver executable (CDTVCDR)
Generated ORACLE request processor main

Two data files for insert test case:

TFILE.FLR

GDFILE.FLR

Complex Mapping Algorithm programs (MDRLX
and MDRLX2)

Verification Module File

Estimated Time for Test : 10 minutes

Special Resource Considerations : Test CDTVC -
Precompilation/Generation of RP-Main must have successfully
completed.

Test Definition

Method of Performing Test : Test case number 1 (enter 01)
must be run first and test case number 11 (enter 11) must be run
last. Test cases numbers 9 and 10 must be run in order (first 9
then 10). All other test cases can be run in any
order.

Expected Test Results : Test results will be shown
below for each of the eleven test cases.

Successful Completion
Criteria for Test

: \$ RUN CDTVCDRZ

CDTVC01 (01):

ENTER CDM RELEASE 2.3 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

01

Result of this test will be population of the TEAM and PLAYER
databases.

CDTVC02 (02):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

02

-----SEARCH COMPLETED-----

AVERAGE: 38767.5714

MINIMUM: 25920.50

MAXIMUM: 50920.50

SUM: 271373.0000

COUNT: 7

CDTVC03 (03):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

03

TEAM NUMBER: 10

PLAYER NAME: BERNIE KOSAR

PLAYER NUMBER: 19

PLAYER SALARY: 50920.500000

TEAM NUMBER: 10

PLAYER NAME: ERNEST BYNER
PLAYER NUMBER: 44
PLAYER SALARY: 46000.000000
TEAM NUMBER: 10
PLAYER NAME: KEVIN MACK
PLAYER NUMBER: 34
PLAYER SALARY: 30500.500000
TEAM NUMBER: 10
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57
PLAYER SALARY: 34750.990000
TEAM NUMBER: 10
PLAYER NAME: CHIP BANKS
PLAYER NUMBER: 56
PLAYER SALARY: 49250.500000
TEAM NUMBER: 10
PLAYER NAME: BOB GOLIC
PLAYER NUMBER: 79
PLAYER SALARY: 25920.500000
TEAM NUMBER: 10
PLAYER NAME: OZZIE NEWSOME
PLAYER NUMBER: 82
PLAYER SALARY: 34030.010000
-----SEARCH COMPLETED-----

CDTVC04 (04):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

UTP620341000
30 September 1990

TEST NUMBER?

04

-----SEARCH COMPLETED-----

The results of this test will be reported on TEAMFILE.DAT, since this is a select into file.

\$ TYPE TEAMFILE.DAT
00000040ASU SUNDEVILS
00000030DAYTON FLYERS
00000020CINCINNATI BEARCATS
00000010CLEVELAND BROWNS

CDTVC05 (05):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

05

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BERNIE KOSAR

PLAYER NUMBER: 19

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CLAY MATHEWS

PLAYER NUMBER: 57

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BOB GOLIC

PLAYER NUMBER: 79

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: KEVIN MACK

PLAYER NUMBER: 34

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: OZZIE NEWSOME
PLAYER NUMBER: 82
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: ERNEST BYNER
PLAYER NUMBER: 44
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: ROMMEL SHORTER
PLAYER NUMBER: 10
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: REGGIE TAYLOR
PLAYER NUMBER: 30
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: ROGER MCCLENDON
PLAYER NUMBER: 21
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: STEVE JACKSON
PLAYER NUMBER: 32
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: JASON STARGILL
PLAYER NUMBER: 25
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: JOE STIFFEND
PLAYER NUMBER: 24
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: DANNY MCCOIN
PLAYER NUMBER: 8
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: ROB NIEHOFF
PLAYER NUMBER: 66

-----SEARCH COMPLETED-----

15 RECORDS SELECTED

CDTVC06 (06):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
06

TEAM NAME: DAYTON FLYERS

TEAM NUMBER: 30

TEAM NAME: ASU SUNDEVILS

TEAM NUMBER: 40

-----SEARCH COMPLETED-----

TOTAL NUMBER OF TEAMS
WITH NO PLAYERS: 2

CDTVC07 (07):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
07

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: KEVIN MACK
PLAYER NUMBER: 34

TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: JOE STIFFEND
PLAYER NUMBER: 24

TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: ROMMEL SHORTER
PLAYER NUMBER: 10

TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: STEVE JACKSON
PLAYER NUMBER: 32

TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: ROB NIEHOFF
PLAYER NUMBER: 66

TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: DANNY MCCOIN
PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

9 RECORDS SELECTED

CDTVC08 (08):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
08

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: BERNIE KOSAR
PLAYER NUMBER: 19
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: BOB GOLIC
PLAYER NUMBER: 79
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CHIP BANKS
PLAYER NUMBER: 56
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: KEVIN MACK
PLAYER NUMBER: 34
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: OZZIE NEWSOME
PLAYER NUMBER: 82
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: ERNEST BYNER
PLAYER NUMBER: 44
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: DANNY MCCOIN
PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

8 RECORDS SELECTED

CDTVC09 (09):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
09

-----MODIFY COMPLETED-----

The result of this test will be an updated database. The attribute GAME_SITE will be updated, and CDTVC10 will select to verify this.

CDTVC10 (10):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
10

NEW GAME SITE: TEMPE, ARIZONA
NEW YARDAGE: 1042.55

-----SEARCH COMPLETED-----

CDTVC11 (11):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
11

The result of this test will be the deletion of all test case data from the TEAM and PLAYER databases.

UTP620341000
30 September 1990

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
00

TESTING COMPLETED.

2.22 CDTVALT - NDDL

Subsystem :
CDM Release: 3.0

Test Name and Number : CDTVALT - NDDL

Objective: This test case redefines the data types for the FORTRAN test cases to follow.

Resource Requirements

Number of terminals : 1
S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request
Supervisor CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2-3 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands:
CDTVALT.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @NDDL CDTVALT

Expected Test Results : The CDM is populated with the information necessary to perform the CDTVTF precompiler test cases.

Successful Completion

Criteria for Test : The execution status will be reported as successful for each NDDL command listed in:

CDTVALT.OUT

2.23 CDTVF - Precompilation/Generation of RP-MAIN

Subsystem :
CDM Release: 3.0

Test Name and Number : CDTVF -
Precompilation/Generation of RP-MAIN

Objective: This test case will precompile ten FORTRAN Application Processes containing NDML requests as a single logical unit of work. It will test the retrieval, insert, modify and delete capabilities of the IISS precompiler.

CDTVF01: CDTVF01 tests the insert action of the precompiler and the generation of calls to a verification module, a conceptual-to-internal complex mapping, and an external-to-conceptual complex mapping. It also tests the capability of several NDML requests in one application process and the capability to insert values from a file as well as values included in the application program.

CDTVF02: CDTVF02 tests the use of the statistical functions AVG, MIN, MAX, SUM, and COUNT.

CDTVF04: CDTVF04 tests the use of SELECT INTO FILE and the use of ORDER BY DESCENDING. It also generates a call to a conceptual-to-external complex mapping.

CDTVF05: CDTVF05 tests the use of parenthetical logic separated with AND and OR operators. It generates calls to conceptual-to-external and internal-to-conceptual complex mappings. It also tests the use of the BETWEEN operator. This NDML retrieval command selects data distributed across two ORACLE databases.

CDTVF06: CDTVF06 tests the use of the Outerjoin (U=) operator with a single NDML query across two ORACLE databases. It also generates a call to a conceptual-to-external complex mapping.

CDTVF07: CDTVF07 tests the use of the query combination command with the UNION operator across two ORACLE databases, also using the BETWEEN operator. It generates calls to a conceptual to external and an internal-to-conceptual complex mapping.

CDTVF08: CDTVF08 tests the use of the XOR operator in a single NDML query across two ORACLE databases. It also generates a conceptual-to-external and an internal-to-conceptual complex mapping.

CDTVF09: CDTVF09 tests the use of the NDML modify command.

CDTVF10: CDTVF10 uses the NDML query capabilities to retrieve the information that was updated in the previous test - CDTVF09.

CDTVF11: CDTVF11 tests the use of several NDML DELETE commands within one application process. One delete will use the BETWEEN operator. It will also generate a call to a conceptual to internal complex mapping.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: CDM File/Module Processing

Capabilities
IISS Precompiler
SQL Request Processor

Generators

Estimated Time for Test : 60 minutes

Special Resource Considerations : This test case requires
that the application driver program be compiled.

\$ @COBGLIB CDTVFD.R.COB

This test case requires ten (10) FORTRAN Application Processes
with NDML requests. The source files for these Application
Processes are: CDTVFD01.PRC, CDTVFD02.PRC, CDTVFD04.PRC,
CDTVFD05.PRC, CDTVFD06.PRC, CDTVFD07.PRC, CDTVFD08.PRC, CDTVFD09.PRC,
CDTVFD10.PRC, and CDTVFD11.PRC. These Application Processes must be
precompiled as a single logical unit of work.

Test Definition

Method of Performing Test : With the ORACLE CDM
database available, at the VAX/VMS prompt (\$) type the test inputs
defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDTV
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: FORTRAN
Enter Language of Source Program(s) (C/ COBOL/FORTRAN): FORTRAN
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDTVDR
Do You Want Obsolete Generated Code Deleted (Y/N)? Y
Does The Application Access Any IBM Databases (Y/N)? N
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV01.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV02.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV04.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV05.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV06.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV07.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV08.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV09.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV10.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTV11.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test will be displayed on the user terminal as indicated above. It will consist of ten (10) modified Application Processes, thirty-nine (39) SQL Request Processor subprograms, sixteen (16) Conceptual/Conceptual Transformer subprograms, seven (7) Conceptual/External Transformer subprograms, and two Request Processor main programs.

Successful Completion
Criteria for Test : All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.24 CDTVF - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : CDTVF - Runtime

Objective: This test case will execute the precompiled FORTRAN Application Processes containing NDML select, insert, modify and delete requests against the ORACLE TEAM and ORACLE PLAYER databases. Ten application processes have been grouped together as a single logical unit of work - one inserts into both databases, seven retrieve data from one or both databases, one modifies the PLAYER database, and one finally deletes from both databases.

Resource Requirements

Number of terminals : 2

S/W Requirements : ORACLE TEAM and PLAYER databases

These databases must have been created using the procedures provided in the "CDM Subsystem Database Build Instructions" Document

NTM: Message and queue server

CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
Application driver executable
(CDTVFDR)

Generated ORACLE request
processor main
Two data files for insert test case:
TFILE.FLR
GDFILF.FLR
Complex Mapping Algorithm Programs
(MDRLX and MDRLX2)
Verification module file

Estimated Time for Test : 10 minutes

Special Resource Considerations : Test CDTVF -
Precompilation/Generation of RP-MAIN must have successfully
completed.

Test Definition

Method of Performing Test : Test case number 1
(enter 01) must be run first and test case number 11 (enter 11)
must be run last. Test cases numbers 9 and 10 must be run in
order (first 9 then 10). All other test cases can be run in any
order.

Expected Test Results : Test results will be shown
below for each of the ten test cases.

Successful Completion
Criteria for Test

: \$ RUN CDTVFDZ

CDTVF01 (01):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
01

Result of this test will be population of the TEAM and PLAYER
databases.

CDTVF02 (02):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
02

-----SEARCH COMPLETED-----

AVERAGE:	38767.57
MINIMUM:	25920.50
MAXIMUM:	50920.50
SUM:	271373.0
COUNT:	7

CDTVF04 (04):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
04

-----SEARCH COMPLETED-----

The results of this test will be reported on TEAMFILE.DAT, since
this is a select into file.

\$ TYPE TEAMFILE.DAT
000000000004OASU SUNDEVILS
000000000003ODAYTON FLYERS
0000000000020CINCINNATI BEARCATS
0000000000010CLEVELAND BROWNS

CDTVF05 (05):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
05

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BERNIE KOSAR

PLAYER NUMBER: 19

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CLAY MATHEWS

PLAYER NUMBER: 57

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BOB GOLIC

PLAYER NUMBER: 79

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: KEVIN MACK

PLAYER NUMBER: 34

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: OZZIE NEWSOME

PLAYER NUMBER: 82

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

UTP620341000
30 September 1990

TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROMMEL SHORTER
PLAYER NUMBER:	10
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	REGGIE TAYLOR
PLAYER NUMBER:	30
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROGER MCCLENDON
PLAYER NUMBER:	21
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	STEVE JACKSON
PLAYER NUMBER:	32
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	JASON STARGILL
PLAYER NUMBER:	25
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	JOE STIFFEND
PLAYER NUMBER:	24
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	DANNY MCCOIN
PLAYER NUMBER:	8
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROB NIEHOFF
PLAYER NUMBER:	66

-----SEARCH COMPLETED-----

15 RECORDS SELECTED

CDTVF06 (06):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
06

TEAM NAME: DAYTON FLYERS

TEAM NUMBER: 30

TEAM NAME: ASU SUNDEVILS

TEAM NUMBER: 40

-----SEARCH COMPLETED-----

TOTAL NUMBER OF TEAMS
WITH NO PLAYERS: 2

CDTVF07 (07):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
07

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CLAY MATHEWS

PLAYER NUMBER: 57

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: KEVIN MACK

PLAYER NUMBER: 34

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: JOE STIFFEND

PLAYER NUMBER: 24

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: ROMMEL SHORTER

PLAYER NUMBER: 10

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: STEVE JACKSON

PLAYER NUMBER: 32

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: ROB NIEHOFF

PLAYER NUMBER: 66

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: DANNY MCCOIN

PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

9 RECORDS SELECTED

CDTVF08 (08):

ENTER CDM RELEASE 2.3 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
08

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BERNIE KOSAR

PLAYER NUMBER: 19

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: BOB GOLIC
PLAYER NUMBER: 79
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CHIP BANKS
PLAYER NUMBER: 56
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: KEVIN MACK
PLAYER NUMBER: 34
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: OZZIE NEWSOME
PLAYER NUMBER: 82
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: ERNEST BYNER
PLAYER NUMBER: 44
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: DANNY MCCOIN
PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

8 RECORDS SELECTED

CDTVF09 (09):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11
TEST NUMBER?
09

-----MODIFY COMPLETED-----

The result of this test will be an updated database. The attribute GAME_SITE will be updated, and CDTVf10 will select to verify this.

CDTVf10 (10):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
10

NEW GAME SITE: TEMPE, ARIZONA

NEW YARDAGE: 1042.55000000

-----SEARCH COMPLETED-----

CDTVf11 (11):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
11

The result of this test will be the deletion of all test case data from the TEAM and PLAYER databases.

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
00

TESTING COMPLETED.

2.25 CDTVDRP - NDDL DROP

Subsystem : CDM Release:
3.0

Test Name and Number : CDTVDRP - NDDL DROP

Objective: This test case executes NDDL Drop commands to drop all the meta data from the CDM used in the previous CDTV and CDTV precompile and runtime test cases.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request

Supervisor CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 10 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands:

CDTVDRP.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CDTVDRP

Expected Test Results : Meta data is dropped from the CDM, ending the CDTV and CDTV test cases.

Successful Completion
Criteria for Test : The execution status will be reported as successful for each NDDL command listed in:
CDTVDRP.OUT

2.26 CDTICRT - NDDL

Subsystem : CDM Release: 3.0

Test Name and Number : CDTICRT - NDDL

Objective: This test case executes NDDL Conceptual, External, Internal, Map, and Complex Map commands to populate the CDM with meta data required for the CDTIC (IBM COBOL) series of precompile and runtime test cases.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request

Supervisor CDM File/Module Processing
Capabilities
NDDL Command Processors

Estimated Time for Test : 35-40 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands:

CDTICRT.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @NDDL CDTICRT

Expected Test Results : The CDM is populated with the information necessary to perform the CDTIC precompiler test cases.

Successful Completion
Criteria for Test : The execution status will be reported as successful for each NDDL command listed in:

CDTICRT.OUT

2.27 CDTIC - Precompilation/Generation of RP-Main

Subsystem
3.0

: CDM

Release:

Test Name and Number : CDTIC -
Precompilation/Generation of RP-MAIN

Objective: This test case consists of eleven Application Processes containing NDML requests as a single logical unit of work. It will test the retrieval, insert, modify and delete capabilities of the IISS precompiler. These processes go against VAX/ORACLE and IBM/DB2 databases. The test case will precompile, compile, generate request processor main programs and link the applications.

CDTIC01: CDTIC01 tests the insert action of the precompiler and the generation of calls to a verification module, a conceptual-to-internal complex mapping, and an external-to-conceptual complex mapping. It also tests the capability of several NDML requests in one application process and the capability to insert values from a file as well as values included in the application program.

CDTIC02: CDTIC02 tests the use of the statistical functions AVG, MIN, MAX, SUM, and COUNT.

CDTIC03: CDTIC03 tests the use of SELECT INTO STRUCTURE and the use of ALL. It also generates a call to an internal-to-conceptual complex mapping.

CDTIC04: CDTIC04 tests the use of SELECT INTO FILE and the use of ORDER BY DESCENDING. It also generates a call to a conceptual-to-external complex mapping.

CDTIC05: CDTIC05 tests the use of parenthetical logic separated with AND and OR operators. It generates calls to conceptual-to-external and internal-to-conceptual complex mappings. It also tests the use of the BETWEEN operator. This NDML retrieval command selects data distributed across two databases, ORACLE and DB2.

CDTIC06: CDTIC06 tests the use of the Outerjoin (U=) operator with a single NDML query across two databases, ORACLE and DB2. It also generates a call to a conceptual-to-external complex mapping.

CDTIC07: CDTIC07 tests the use of the query combination command with the UNION operator across two databases, ORACLE and DB2, also using the BETWEEN operator. It generates calls to a conceptual-to-external and an internal-to-conceptual complex mapping.

CDTIC08: CDTIC08 tests the use of the XOR operator in a single NDML query across two databases, ORACLE and DB2. It also generates a conceptual-to-external and an internal-to-conceptual complex mapping.

CDTIC09: CDTIC09 tests the use of the NDML modify command.

CDTIC10: CDTIC10 uses the NDML query capabilities to retrieve the information that was updated in the previous test - CDTIC09.

CDTIC11: CDTIC11 tests the use of several NDML DELETE commands within one application process. One delete will use the BETWEEN operator. It also generates a call to a conceptual-to-internal complex mapping.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM Database

CDMP: CDM File/Module Processing

Capabilities
IISS Precompiler
SQL Request Processor

Generators

Remote Job Entry (SNARJE) from VAX to IBM

Estimated Time for Test : 60 minutes

Special Resource Considerations : This test case requires that the two complex mapping algorithm programs and the application driver program be compiled:

\$@ COBGLIB MDRLX.COB
\$@ COBGLIB MDRLX2.COB
\$@ COBGLIB CDTVCDR.COB

This test case requires eleven (11) Application Processes with NDML requests. The source files for these Application Processes are: CDTIC01.PRC, CDTIC02.PRC, CDTIC03.PRC, CDTIC04.PRC, CDTIC05.PRC, CDTIC06.PRC, CDTIC07.PRC, CDTIC08.PRC, CDTIC09.PRC, CDTIC10.PRC, and CDTIC11.PRC. These Application Processes must be precompiled as a single logical unit of work.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDTIC
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated request Processor: COBOL
Enter Language of Source Program(s) (C/C)B) L/FORTRAN): COBOL
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDTICDR
Do You Want Obsolete Generated Code Deleted (Y/N)? : Y
Does The Application Access Any IBM Databases (Y/N)? Y
Enter your IBM Username: (Dependent on your testing environment)
Enter your IBM Password: (Dependent on your testing environment)
Enter the PDS name for your IBM source code: (Dependent on your testing environment)
Enter Name Of PRC (C/R To Stop, Include Extension): DTIC01.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC02.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC03.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC04.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC05.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC06.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC07.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC08.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC09.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC10.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIC11.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test will be displayed on the user terminal as indicated above. It will consist of eleven (11) modified Application Processes, forty (40) SQL Request Processor subprograms, sixteen (16) Conceptual/Conceptual Transformer subprograms, eight (8) Conceptual/External Transformer subprograms, and two Request Processor main programs.

Successful Completion
Criteria for Test : All of the above generated programs and subprograms will compile free of warning and fatal errors on the designated host (VAX or IBM).

2.28 CDTIC - Runtime

Subsystem : CDM Release:
3.0

Test Name and Number : CDTIC - Runtime

Objective: This test case will execute the precompiled COBOL Application Processes containing NDML select, insert, modify and delete requests against the ORACLE TEAM and DB2 CDCJA databases. Eleven application processes have been grouped together as a single logical unit of work - one inserts into both databases, eight retrieve data from one or both databases, one modifies the CDCJA database, and one finally deletes from both databases. A driver program is also used.

Resource Requirements

Number of terminals : 2

S/W Requirements : ORACLE TEAM and DB2 CDCJA
databases

These database must have been created using the procedures provided in the "CDM Subsystem Database Build Instructions" Document

NTM: Message and queue server
CDMP: Distributed Request Supervisor
CDM File/Module Processing

executable

Capabilities
Application driver

(CDTICDR)
Generated ORACLE request
processor main
Generated DB2 request
processor main

Two data files for insert test case:

TFILE.FLR
GDFILE.FLR

Complex Mapping Algorithm programs (MDRLX and MDRLX2)
Verification Module File

Estimated Time for Test : 10 minutes

Special Resource Considerations : Test CDTIC -
Precompilation/Generation of RP-Main must have successfully
completed.

Test Definition

Method of Performing Test : Test case number 1
(enter 01) must be run first and test case number 11 (enter 11)
must be run last. Test cases numbers 9 and 10 must be run in
order (first 9 then 10). All other test cases can be run in any
order.

Expected Test Results : Test results will be shown below
for each of the eleven test cases.

Successful Completion
Criteria for Test : \$ RUN CDTICDRZ

CDTIC01 (01):

ENTER CDM RELEASE 2.3 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
01

Result of this test will be population of the TEAM and
CDCJA databases.

CDTIC02 (02):

ENTER CDM RELEASE 2.3 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
02

-----SEARCH COMPLETED-----

AVERAGE:	38767.5714
MINIMUM:	25920.5000
MAXIMUM:	50920.5000
SUM:	271373.0000
COUNT:	7

CDTIC03 (03):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
03

UTP620341000
30 September 1990

TEAM NUMBER:	10
PLAYER NAME:	OZZIE NEWSOME
PLAYER NUMBER:	82
PLAYER SALARY:	34030.010000
TEAM NUMBER:	10
PLAYER NAME:	BOB GOLIC
PLAYER NUMBER:	79
PLAYER SALARY:	25920.500000
TEAM NUMBER:	10
PLAYER NAME:	CHIP BANKS
PLAYER NUMBER:	56
PLAYER SALARY:	49250.500000
TEAM NUMBER:	10
PLAYER NAME:	CLAY MATHEWS
PLAYER NUMBER:	57
PLAYER SALARY:	34750.990000
TEAM NUMBER:	10
PLAYER NAME:	KEVIN MACK
PLAYER NUMBER:	34
PLAYER SALARY:	30500.500000
TEAM NUMBER:	10
PLAYER NAME:	ERNEST BYNER
PLAYER NUMBER:	44
PLAYER SALARY:	46000.000000
TEAM NUMBER:	10
PLAYER NAME:	BERNIE KOSAR
PLAYER NUMBER:	19
PLAYER SALARY:	50920.500000

-----SEARCH COMPLETED-----

CDTIC04 (04):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

04

-----SEARCH COMPLETED-----

The results of this test will be reported on TEAMFILE.DAT, since this is a select into file.

\$ TYPE TEAMFILE.DAT
00000040ASU SUNDEVILS
00000030DAYTON FLYERS
00000020CINCINNATI BEARCATS
00000010CLEVELAND BROWNS

CDTIC05 (05):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

05

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: OZZIE NEWSOME

PLAYER NUMBER: 82

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BOB GOLIC

PLAYER NUMBER: 79

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME:	BERNIE KOSAR
PLAYER NUMBER:	19
TEAM NAME:	CLEVELAND BROWNS
PLAYER NAME:	KEVIN MACK
PLAYER NUMBER:	34
TEAM NAME:	CLEVELAND BROWNS
PLAYER NAME:	CLAY MATHEWS
PLAYER NUMBER:	57
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	DANNY MCCOIN
PLAYER NUMBER:	8
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROMMEL SHORTER
PLAYER NUMBER:	10
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROGER MCCLENDON
PLAYER NUMBER:	21
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	STEVE JACKSON
PLAYER NUMBER:	32
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	REGGIE TAYLOR
PLAYER NUMBER:	30
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	JOE STIFFEND
PLAYER NUMBER:	24
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROB NIEHOFF
PLAYER NUMBER:	66
TEAM NAME:	CINCINNATI BEARCATS

PLAYER NAME: JASON STARGILL

PLAYER NUMBER: 25

-----SEARCH COMPLETED-----

15 RECORDS SELECTED

CDTIC06 (06):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
06

TEAM NAME: DAYTONFLYERS

TEAM NUMBER: 30

TEAM NAME: ASU SUNDEVILS

TEAM NUMBER: 40

-----SEARCH COMPLETED-----

TOTAL NUMBER OF TEAMS
WITH NO PLAYERS: 2

CDTIC07 (07):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

07

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: DANNY MCCOIN

PLAYER NUMBER: 8

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: ROMMEL SHORTER

PLAYER NUMBER: 10

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: JOE STIFFEND

PLAYER NUMBER: 24

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: STEVE JACKSON

PLAYER NUMBER: 32

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: ROB NIEHOFF

PLAYER NUMBER: 66

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: KEVIN MACK

PLAYER NUMBER: 4

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CHIP BANKS
PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57

-----SEARCH COMPLETED-----

9 RECORDS SELECTED

CDTIC08 (08):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
08

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: OZZIE NEWSOME
PLAYER NUMBER: 82

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CHIP BANKS
PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: BOB GOLIC
PLAYER NUMBER: 79

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: ERNEST BYNER
PLAYER NUMBER: 44

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: BERNIE KOSAR
PLAYER NUMBER: 19

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: KEVIN MACK
PLAYER NUMBER: 34

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57

TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: DANNY MCCOIN
PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

8 RECORDS SELECTED

CDTIC09 (09):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
09

-----MODIFY COMPLETED-----

The result of this test will be an updated database. The attribute GAME_SITE will be updated, and CDTIC10 will select to verify this.

CDTIC10 (10):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
10

NEW GAME SITE: TEMPE, ARIZONA

NEW YARDAGE: 1042.55

-----SEARCH COMPLETED-----

CDTIC11 (11):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
11

The result of this test will be the deletion of all test case data
from the TZAM and CDCJA databases.

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 03, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
00

TESTING COMPLETED.

2.29 CDTIALT - NDDL

Subsystem : CDM Release:
3.0

Test Name and Number : CDTIALT - NDDL

Objective: This test case redefines the data types for the
FORTRAN test cases to follow.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request

Supervisor CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2-3 minutes

Special Resource Considerations : This test case requires
the following data file which contains the NDDL commands:

CDTIALT.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM
database available, at the VAX/VMS prompt (\$) type the test input
defined below.

Test Inputs : \$ @NDDL CDTIALT

Expected Test Results : The CDM is populated with the
information necessary to perform the CDTIF precompiler test
cases.

Successful Completion Criteria for Test : The
execution status will be reported as successful for each NDDL
command listed in:

CDTIALT.OUT

2.30 CDTIF - Precompilation/Generation of RP-MAIN

Subsystem
3.0

: CDM

Release:

Test Name and Number : CDTIF -
Precompilation/Generation of RP-MAIN

Objective: This test case will precompile ten FORTRAN Application Processes containing NDML requests as a single logical unit of work. It will test the retrieval, insert, modify and delete capabilities of the IISS precompiler. These processes go against two IBM/DP2 databases.

CDTIF01: CDTIF01 tests the insert action of the precompiler and the generation of calls to a verification module, a conceptual-to-internal complex mapping, and an external-to-conceptual complex mapping. It also tests the capability of several NDML requests in one application process and the capability to insert values from a file as well as values included in the application program.

CDTIF02: CDTIF02 tests the use of the statistical functions AVG, MIN, MAX, SUM, and COUNT.

CDTIF04: CDTIF04 tests the use of SELECT INTO FILE and the use of ORDER BY DESCENDING. It also generates a call to a conceptual-to-external complex mapping.

CDTIF05: CDTIF05 tests the use of parenthetical logic separated with AND and OR operators. It generates calls to conceptual-to-external and internal-to-conceptual complex mappings. It also tests the use of the BETWEEN operator. This NDML retrieval command selects data distributed across an ORACLE and a DB2 database.

CDTIF06: CDTIF06 tests the use of the Outerjoin (U=) operator with a single NDML query across an ORACLE and DB2 database. It also generates a call to a conceptual-to-external complex mapping.

CDTIF07: CDTIF07 tests the use of the query combination command with the UNION operator across an ORACLE and a DB2 database, also using the BETWEEN operator. It generates calls to a conceptual to external and an internal-to-conceptual complex mapping.

CDTIF08: CDTIF08 tests the use of the XOR operator in a single NDML query across an ORACLE and a DB2 database. It also generates a conceptual-to-external and an internal-to- conceptual complex mapping.

CDTIF09: CDTIF09 tests the use of the NDML modify command.

CDTIF10: CDTIF10 uses the NDML query capabilities to retrieve the information that was updated in the previous test - CDTIF09.

CDTIF11: CDTIF11 tests the use of several NDML DELETE commands within one application process. One delete will use the BETWEEN operator. It will also generate a call to a conceptual to internal complex mapping.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: CDM File/Module Processing

Capabilities
IISS Precompiler
SQL Request Processor

Generators

Remote Job Entry (SNARJE) from
VAX to IBM
Estimated Time for Test : 60 minutes

Special Resource Considerations : This test case requires that the application driver program be compiled.

\$ @COBGLIB CDTIFDR.COB

This test case requires ten (10) FORTRAN Application Processes with NDML requests. The source files for these Application Processes are: CDTIF01.PRC, CDTIF02.PRC, CDTIF04.PRC, CDTIF05.PRC, CDTIF06.PRC, CDTIF07.PRC, CDTIF08.PRC, CDTIF09.PRC, CDTIF10.PRC, and CDTIF11.PRC. These Application Processes must be precompiled as a single logical unit of work.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

Enter Name Of Logical Unit Of Work: CDTIF
Enter Name Of Host Where Application Will Run: VAX
Enter Desired Language of Generated Request Processors: FORTRAN
Enter Language of Source Program(s) (C/COBOL/FORTRAN): FORTRAN
Enter the type of Embedded Language Used (NDML/SQL): NDML
Enter Your CDM Username/Password: CDM/CDM
Enter Module Name Of Your Application: CDTIFDR
Do You Want Obsolete Generated Code Deleted (Y/N)? Y
Does The Application Access Any IBM Databases (Y/N)? Y
Enter your IBM Username: (Dependent on your testing environment)
Enter your IBM Password: (Dependent on your testing environment)
Enter the PDS name for your IBM source code: (Dependent on your testing environment)
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF0.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF02.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF04.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF05.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF06.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF07.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF08.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF09.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF10.PRC
Enter Name Of PRC (C/R To Stop, Include Extension): CDTIF11.PRC
Enter Name Of PRC (C/R To Stop, Include Extension):

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test will be displayed on the user terminal as indicated above. It will consist of ten (10) modified Application Processes, thirty-nine (39) SQL Request Processor subprograms, sixteen (16) Conceptual/Conceptual Transformer subprograms, seven (7) Conceptual/External Transformer subprograms, and two Request Processor main programs.

Successful Completion
Criteria for Test : All of the above generated programs and subprograms will compile free of warning and fatal errors on the designated host (VAX or IBM).

2.31 CDTIF - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number : CDTIF - Runtime

Objective: This test case will execute the precompiled FORTRAN Application Processes containing NDML select, insert, modify and delete requests against the ORACLE TEAM and DB2 CDCJA databases. Ten application processes have been grouped together as a single logical unit of work - one inserts into both databases, seven retrieve data from one or both databases, one modifies the CDCJA database, and one finally deletes from both databases.

Resource Requirements

Number of terminals : 2

S/W Requirements : DB2 CDCJA and CDCSG databases

These databases must have been created using the procedures provided in the "CDM Subsystem Database Build Instructions" Document

NTM: Message and queue server

CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities

Application driver executable (CDTIFDR)

Generated DB2 request processor mains

Two data files for insert test case:

TFILF.DAT

GDFILF.DAT

Complex Mapping Algorithm Programs (MDRLX and MDRLX2)

Estimated Time for Test : 10 minutes

Special Resource Considerations:

Test CDTIF - Precompilation/Generation of RP-MAIN must have successfully completed.

Test Definition

Method of Performing Test:

Test case number 1 (enter 01) must be run first and test case number 11 (enter 11) must be run last. Test cases numbers 9 and 10 must be run in order (first 9 then 10). All other test cases can be run in any order.

Expected Test Results: Test results will be shown below for each of the eleven test cases.

Successful Completion
Criteria for Test

: \$ RUN CDTIFDRZ

CDTIF01 (01):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

01

Result of this test will be population of the CDCSG and CDCJA
databases.

CDTIF02 (02):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

02

-----SEARCH COMPLETED-----

AVERAGE: 38767.57

MINIMUM: 25920.50

MAXIMUM: 50920.50

SUM: 271373.00

COUNT: 7

CDTIF04 (04):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

04

-----SEARCH COMPLETED-----

The results of this test will be reported on TEAMFILE.DAT, since
this is a select into file.

UTP620341000
30 September 1990

\$ TYPE TEAMFILE.DAT
000000000040ASU SUNDEVILS
000000000030DAYTON FLYERS
000000000020CINCINNATI BEARCATS
000000000010CLEVELAND BROWNS

CDTIF05 (05):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
05

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BERNIE KOSAR

PLAYER NUMBER: 19

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CLAY MATHEWS

PLAYER NUMBER: 57

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BOB GOLIC

PLAYER NUMBER: 79

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: KEVIN MACK

PLAYER NUMBER: 34

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: OZZIE NEWSOME

PLAYER NUMBER: 82

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROMMEL SHORTER
PLAYER NUMBER:	10
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	REGGIE TAYLOR
PLAYER NUMBER:	30
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROGER MCCLENDON
PLAYER NUMBER:	21
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	STEVE JACKSON
PLAYER NUMBER:	32
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	JASON STARGILL
PLAYER NUMBER:	25
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	JOE STIFFEND
PLAYER NUMBER:	24
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	DANNY MCCOIN
PLAYER NUMBER:	8
TEAM NAME:	CINCINNATI BEARCATS
PLAYER NAME:	ROB NIEHOFF
PLAYER NUMBER:	66

-----SEARCH COMPLETED-----

15 RECORDS SELECTED

CDTIF06 (06):

UTP620341000
30 September 1990

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
06

TEAM NAME: DAYTON FLYERS

TEAM NUMBER: 30

TEAM NAME: ASU SUNDEVILS

TEAM NUMBER: 40

-----SEARCH COMPLETED-----

TOTAL NUMBER OF TEAMS
WITH NO PLAYERS: 2

CDTIF07 (07):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
07

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: ERNEST BYNER

PLAYER NUMBER: 44

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CHIP BANKS

PLAYER NUMBER: 56

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: CLAY MATHEWS

PLAYER NUMBER: 57

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: KEVIN MACK

PLAYER NUMBER: 34

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: JOE STIFFEND

PLAYER NUMBER: 24

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: ROMMEL SHORTER

PLAYER NUMBER: 10

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: STEVE JACKSON

PLAYER NUMBER: 32

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: ROB NIEHOFF

PLAYER NUMBER: 66

TEAM NAME: CINCINNATI BEARCATS

PLAYER NAME: DANNY MCCOIN

PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

9 RECORDS SELECTED

CDTIF08 (08):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

08

TEAM NAME: CLEVELAND BROWNS

PLAYER NAME: BERNIE KOSAR

PLAYER NUMBER: 19

UTP620341000
30 September 1990

TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CLAY MATHEWS
PLAYER NUMBER: 57
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: BOB GOLIC
PLAYER NUMBER: 79
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: CHIP BANKS
PLAYER NUMBER: 56
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: KEVIN MACK
PLAYER NUMBER: 34
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: OZZIE NEWSOME
PLAYER NUMBER: 82
TEAM NAME: CLEVELAND BROWNS
PLAYER NAME: ERNEST BYNER
PLAYER NUMBER: 44
TEAM NAME: CINCINNATI BEARCATS
PLAYER NAME: DANNY MCCOIN
PLAYER NUMBER: 8

-----SEARCH COMPLETED-----

8 RECORDS SELECTED

CDTIF09 (09):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?

09

-----MODIFY COMPLETED-----

The result of this test will be an updated database. The attribute GAME_SITE will be updated, and CDTVf10 will select to verify this.

CDTIF10 (10):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
10

NEW GAME SITE: TEMPE, ARIZONA

NEW YARDAGE: 1042.55000000

-----SEARCH COMPLETED-----

CDTIF11 (11):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
11

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE: 01, 02, 04, 05, 06,
07, 08, 09, 10, 11

TEST NUMBER?
00

TESTING COMPLETED.

The result of this test will be the deletion of all test case data from the CDCSG and CDCJA atabases.

2.32 CDTIDRP - NDDL DROP

Subsystem : CDM
3.0

Release:

Test Name and Number : CDTIDRP - NDDL DROP

Objective: This test case executes NDDL Drop commands to drop all the meta data from the CDM used in the previous CDTIC and CDTIF precompile and runtime test cases.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request

Supervisor CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 10 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands:
CDTIDRP.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CDTIDRP

Expected Test Results : Meta data is dropped from the CDM, ending the CDTVIC and CDTIF test cases.

Successful Completion
Criteria for Test : The execution status will be reported as successful for each NDDL command listed in:

CDTIDRP.OUT

2.33 UNIVCRT - NDDL

Subsystem : CDM Release:
3.0

Test Name and Number : UNIVCRT - NDDL

Objective: This test case executes NDDL Conceptual, External, Internal, and Map commands to populate the CDM with meta data required for the CDUNI series of precompile and runtime test cases. The scope of this test case is a small University model. The conceptual schema consists of nine entities. These entities map to two databases - a VAX-11 database with four records and an ORACLE database with eight records. Nine simple views and eight complex views have been created for the following applications.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
ORACLE UNIV1 and VAX-11 UNIV2
databases

These databases must have been created using the procedures in the

Build "CDM Subsystem Database
Instructions" Document
Supervisor, CDMP: Distributed Request
CDM File/Module Processing
Capabilities
NDDL Command Processors.

Estimated Time for Test : 35-40 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands: This file must be edited to specify the exact location of the VAX-11 database.

UNIVCRT.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs : \$ @NDDL UNIVCRT

UTP620341000
30 September 1990

Expected Test Results : The CDM is populated with the information necessary to perform the CDUNI precompiler test cases.

Successful Completion
Criteria for Test : The execution status will be reported as successful for each NDDL command listed in:

UNIVCRT.OUT

2.34 CDUNI - Precompilation/Generate RP-MAIN

Subsystem: CDM

Release: 3.0

Test Name and Number: CDUNI - Precompilation/Generate RP-MAIN

Objective: This series of test cases demonstrates the functionality of TASK 8005 and TASK 8016 for Release 3.0.

The test cases demonstrate:

- o Retrieval and update of a single entity view with restricted rows.
- o Retrieval from multiple entity views that have row-restriction criteria, AND/OR/XOR logic, parenthesized logic, row-column comparison, use arithmetic operators, etc.
- o Restricted retrieval and update when specified through use of the "disallow" clause.
- o Non-guaranteed update when the "allow" update clause is specified.
- o Finally, a combination of the above.

This test case will precompile eleven application processes containing NDML requests as a single logical unit of work.

The first four test cases test the non-guaranteed distributed update capabilities while the next seven demonstrate the distributed retrieval and enhanced view specifications. The objectives of each individual test case are listed below:

CDUNI01: INSERT into the entity named COURSE which is mapped to two records for separate preferences. It will test the "ALLOW UPDATE" capability. Two insert routines will be generated to insert into two databases.

CDUNI02: DELETE from entity named COURSE, qualifying on a specific course code. Since the entity is mapped to two records and update has been allowed, this will result in two subtransactions. One transaction deletes from the database UNIV1 and the second transaction results in a modify to database UNIV2 because of unmapped datafields.

CDUNI03: MODIFY entity OFFERING, if certain qualifications are met. Since this entity is mapped to two records and the ALLOW UPDATE is in effect, this will result in two modify subtransactions to the two databases.

CDUNI05: DELETE from entity INSTRUCTOR. This entity maps to two databases for different preferences and has the "ALLOW UPDATE" clause specified. One transaction deletes from database UNIV1 and the second transaction results in a modify to database UNIV2. The WHERE clause in the delete statement contains parenthesized and "OR" logic.

CDUNI06: This application SELECTs information from two views. One view is a list of all instructors who teach computer science (CS) in the spring or fall semesters OR teach non CS courses in the summer semester. This view contains an exclusive-or (XOR) condition.

CDUNI07: MODIFY entity OFFERING using the view NO_CREDIT_HOURS. This is a simple entity view which has qualifying criteria - it contains only computer science (CS) courses which are currently assigned no (null) credit hours. Even though the user coded program has no WHERE clause in the NDML statement, the qualification criteria of the view WHERE clause is automatically appended to the NDML statement.

CDUNI08: INSERT into a single entity view A_TO_M_STUDENT. This view has been built with qualification criteria, but these qualifications will have no effect on the NDML statement. This application is inserting a student with a name that is not alphabetically between A and M (i.e. not meeting the WHERE clause criteria), yet the INSERT is successful.

CDUNI09: SELECT from a two-entity view which contains parenthesized logic and a BETWEEN clause. The application presents a list of all computer courses offered in summer. Both entities have the "ALLOW RETRIEVAL" clause specified. Since both entities map to both databases for different preferences, only the first preference mapping will be selected.

CDUNI10: SELECT from a two-entity view which contains an OUTER JOIN qualification in the view WHERE clause. This will present a list of all students not currently enrolled in any class section.

CDUNI11: SELECT from a two entity view, ACTIVE_STUDENT, which will present a list of all students currently enrolled in a class. The list will be outputted with duplicates removed (the distinct clause is specified during the view creation phase) and in alphabetical order.

CDUNI12: This test case SELECTs information from two views - OVERPAID-UNDERQUALIFIED and CLASS-SECTIONS-GROWING. As the view names suggest, these views are multi-entity views that have a host of qualification criteria.

Resource Requirements

Number of terminals	: 1
S/W Requirements	: ORACLE CDM DATABASE CDMP: CDM File/Module Processing Capabilities IISS Precompiler SQL Request Processor Generators
Estimated Time for Test	: 60 minutes

Special Resource Considerations : This test case requires that the application driver program be compiled:

\$@COBGLIB CDTVCDR.COB

This test case requires eleven (11) Application Processes with NDML requests. The source files for these Application Processes are: CDUNI01.PRC, CDUNI02.PRC, CDUNI03.PRC, CDUNI05.PRC, CDUNI06.PRC, CDUNI07.PRC, CDUNI08.PRC, CDUNI09.PRC, CDUNI10.PRC, CDUNI11.PRC and CDUNI12.PRC. These Application Processes must be precompiled, compiled, and linked as a single logical unit of work using the procedure file GENAP.COM.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @GENAP

APPLICATION GENERATOR MAIN MENU

1. Gap Only
2. Precompile Only
3. Link Only
4. Gap And Precompile
5. Precompile And Link
6. Gap, Precompile, And Link
7. Help - Option Descriptions
8. Exit

PLEASE ENTER AN OPTION NUMBER: 5

ENTER NAME OF THE LOGICAL UNIT OF WORK: CDUNI
ENTER NAME OF HOST WHERE APPLICATION WILL RUN: VAX
ENTER DESIRED LANGUAGE OF GENERATED REQUEST PROCESSORS: COBOL
ENTER LANGUAGE OF SOURCE PROGRAMS(C/COBOL/FORTRAN): COBOL
ENTER THE TYPE OF EMBEDDED LANGUAGE USED (NDML/SQL): NDML
ENTER YOUR CDM USERNAME/PASSWORD: CDM/CDM
ENTER MODULE NAME OF YOUR APPLICATION: CDUNIDR
DO YOU WANT OBSOLETE GENERATED CODE DELETED (Y/N)? : Y
DOES THE APPLICATION ACCESS ANY IBM DATABASES (Y/N)? N
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI01.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI02.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI03.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI05.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI06.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI07.PRC

ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI08.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI09.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI10.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION):
CDUNI11.PRC
ENTER NAME OF PRC FILE (C/R TO STOP, INCLUDE EXTENSION): CDUNI12

NDML PRECOMPILE SUCCESSFULLY COMPLETED
GENERATION OF REQUEST PROCESSOR MAIN(s) COMPLETE
LINKING COMPLETED

Expected Test Results : The results of this test can be found on CDUNI.MSG. It will consist of eleven (11) modified Application Processes, twenty-nine (29) SQL Request Processor subprograms, six (6) Conceptual/Conceptual Transformer subprograms, five (5) Conceptual/External Transformer subprograms, and two Request Processor main programs.

Successful Completion
Criteria for Test : All of the above generated programs and subprograms will compile free of warning and fatal errors.

2.35 CDUNI - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number : CDUNI - Runtime

Objective:

This test case will execute the precompiled COBOL Application Processes containing NDML select, insert, modify and delete requests against the ORACLE UNIV1 and VAX-11 UNIV2 databases. Twelve application processes have been grouped together as a single logical unit of work - one acts as a driver, the others either retrieve or update from one or both databases.

Resource Requirements

Number of terminals : 2

S/W Requirements:

ORACLE UNIV1 and VAX-11 UNIV2 databases. These databases must have been created using the procedures in the "CDM Subsystem Database Build Instructions" Document.

NTM: Message and queue server

CDMP: Distributed Request Supervisor

CDM File/Module Processing Capabilities

Application driver executable CDUNIDR

Estimated Time for Test: 10 minutes

Special Resource Considerations : None

Test Definition

Method of Performing Test : Execute the CDUNIDR.EXE executable.

Expected Test Results:

Test results will be shown below for each of the twelve test cases. The test cases may be run in any order, but for our purposes we will execute them in sequence.

Successful Completion Criteria for Test: \$ RUN CDUNIDR

CDUNI01 (01):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):

ENTER 00 TO EXIT.

LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
07, 08, 09, 10, 11, 12

TEST NUMBER ? 01

INSERTING INTO COURSE

INSERT COMPLETED

CDUNI02 (02):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
 07, 08, 09, 10, 11, 12
TEST NUMBER ? 02
DELETING FROM COURSE
DELETE COMPLETE

CDUNI03 (03):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
 07, 08, 09, 10, 11, 12
TEST NUMBER ? 03
MODIFYING OFFERING
MODIFY COMPLETE

CDUNI05 (05):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
 07, 08, 09, 10, 11, 12
TEST NUMBER ? 05
DELETING FROM INSTRUCTOR
DELETE COMPLETE

CDUNI06 (06):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
 07, 08, 09, 10, 11, 12

TEST NUMBER ? 06
LIST OF INSTRUCTORS WHO TEACH
CS IN NON-SUMMER or NON-CS IN SUMMER
NAME : MR. INSTRUCTOR
COURSE CODE : CSO
SEMESTER : SUMMER

NAME : MS. INSTRUCTOR
COURSE CODE : CEM
SEMESTER : SUMMER

NAME : MS. INSTRUCTOR
COURSE CODE : ACC
SEMESTER : SUMMER

NAME : PROF CARSON
COURSE CODE : CSO
SEMESTER : SUMMER

NAME : PROF CARSON
COURSE CODE : CSO
SEMESTER : SUMMER

NAME : PROF GUILD
COURSE CODE : CSO

SEMESTER : SUMMER

NAME : PROF SANDERS
COURSE CODE : PHL
SEMESTER : SUMMER

CDUNI07 (07):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
07, 08, 09, 10, 11, 12
TEST NUMBER ? 07
MODIFYING VIEW NO_CREDIT_HOURS
MODIFY COMPLETE

CDUNI08 (08):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
07, 08, 09, 10, 11, 12
TEST NUMBER ? 08
INSERTING INTO VIEW A_TO_M_STUDENT
INSERT COMPLETE

CDUNI09 (09):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
07, 08, 09, 10, 11, 12
TEST NUMBER ? 09
LIST OF COMPUTER COURSES OFFERED IN SUMMER
COURSE CODE : CED
COURSE NAME : INTRO TO SYS DESIGN
CREDIT HOURS : 3

COURSE CODE : CEG
COURSE NAME : INTRO TO COMP ENG
CREDIT HOURS : 4

COURSE CODE : CEM
COURSE NAME : INTRO TO MICROS
CREDIT HOURS : 4

COURSE CODE : CSO
COURSE NAME : INTRO TO OPERAT SYS
CREDIT HOURS : 4

CDUNI10 (10):

ENTER CDM RELEASE 3.0 TEST NUMBER (PIC 99):
ENTER 00 TO EXIT.
LEGITIMATE TESTS ARE : 01, 02, 03, 05, 06
07, 08, 09, 10, 11, 12
TEST NUMBER ? 10
LIST OF STUDENTS NOT ENROLLED IN ANY CLASS
SUSAN

2.36 UNIVDRP - NDDL

Subsystem : CDM Release: 3.0
Test Name and Number : UNIVDRP - NDDL

Objective: This test case executes NDDL Drop commands to drop all the meta data from the CDM that was used in the previous CDUNI precompile and runtime test cases.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE

CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands:

UNIVDRP.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test input defined below.

Test Inputs \$ @NDDL UNIVDRP

Expected Test Results : Meta data is dropped from the CDM, ending the CDUNI test case.

Successful Completion

Criteria for Test : The execution status will be reported as successful for each NDDL command listed in:

UNIVDRP.OUT

2.37 NDDL01 - Runtime

Subsystem :
CDM Release: 3.0

Test Name and Number : NDDL01 - Runtime

Objective: This test case will execute the following conceptual schema commands:

```
CREATE MODEL
CREATE ATTRIBUTE
CREATE ENTITY .. KEY .. OWNED ATTRIBUTE
CREATE RELATION .. MIGRATES
CREATE CATEGORY, CREATE DOMAIN
```

This is the first in the series of test cases NDDL01 through NDDL06.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 3 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands to create the model, entities, attributes and relations:

CSTEST1.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CSTEST1

Expected Test Results : The results of this test case will be the creation of a model, a domain, seven attributes, four entities, two link relations and a category relation.

The CDM tables affected are:

UTP620341000
30 September 1990

MODEL_CLASS	LINK_RELATION
ATTRIBUTE_CLASS	CATEGORY_RELATION
ATTRIBUTE_NAME	CATEGORY_MEMBER
ENTITY_CLASS	COMPLETE_RELATION
ENTITY_NAME	DOMAIN_CLASS
RELATION_CLASS	DOMAIN_VALUE
KEY_CLASS	USER_DEF_DATA_TYPE
KEY_CLASS_MEMBER	
COMPLETE_RELATION	
ATTRIBUTE_USE_CL	
INHERITED_ATT_USE	

Successful Completion

Criteria for Test

: The execution status will be
reported as successful for each command listed in CTEST1.OUT.

2.38 NDDL02 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL02 - Runtime

Objective: This test case will execute the following conceptual schema commands:

```
ALTER MODEL
ALTER ENTITY .. ADD KEY
ALTER RELATION .. ADD MIGRATES
ALTER CATEGORY...ADD CATEGORY
```

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities

NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands to alter the entity and relation definitions:

CSTEST2.DAT

In addition, test case NDDL01 should have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CSTEST2

Expected Test Results : The results of this test case will be an entity with a new key class, and a relation class migrating this key class, and a new category entity added to the category relation.

Successful Completion

Criteria for Test : The execution status will be reported as successful for each command listed in CSTEST2.OUT.

2.39 NDDL03 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL03 - Runtime

Objective: This test case executes the following NDDL commands:

```
CREATE ALIAS
ALTER ATTRIBUTE .. ADD KEYWORD
ALTER ATTRIBUTE .. ADD DOMAIN
DESCRIBE
```

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource

Considerations : This test case requires the following data file which contains the NDDL commands:

CSTEST3.DAT.

In addition, test cases NDDL01 thru NDDL02 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CSTEST3

Expected Test Results : The results of this test case will be: Description text for an attribute, link relation and category relation. A key word specified for an attribute. A domain for the attribute.

Successful Completion

Criteria for Test : The execution status will be reported as successful for each command listed in CSTEST3.OUT.

2.40 NDDL04 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL04 - Runtime

Objective: This test case executes the following NDDL commands:

ALTER ALIAS
DROP ALIAS
ALTER ENTITY .. DROP ATTRIBUTE
DROP KEYWORD

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands: CSTEST4.DAT

Test Definition

Method of Performing Test: With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CSTEST4

Expected Test Results: The results of this test case will be:

- . Alias name created for an entity
- . Alias name dropped for an attribute
- . An attribute deleted from an entity and a keyword dropped from an attribute
- . The keyword itself deleted

Successful Completion Criteria for Test: The execution status will be reported as successful for each command listed in CSTEST4.OUT

2.41 NDDL05 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL05 - Runtime

Objective: This test cast executes the following NDDL conceptual schema commands to drop a specified entity, an attribute and a relation:

DROP RELATION
DROP ATTRIBUTE
DROP ENTITY
DROP CATEGORY

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL commands: CSTEST5.DAT

In addition, test cases NDDL01 thru NDDL04 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CSTEST5

Expected Test Results : The test case will result in an attribute, link relation, category relation, and entity being dropped from the model.

Successful Completion Criteria for Test: The execution status will be listed as successful for each command listed in CSTEST5.OUT.

2.42 NDDL06 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL06 - Runtime

Objective: This test case will execute the following NDDL command to delete the model created and manipulated in test cases NDDL01 thru NDDL05:

DROP MODEL

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the following data file which contains the NDDL command: CTEST6.DAT

In addition, test cases NDDL01 thru NDDL05 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CTEST6

Expected Test Results : This test case will drop a model. All CDM conceptual schema tables will be affected.

Successful Completion Criteria for Test: The execution status will be listed as successful for each command listed in CTEST6.OUT.

2.43 NDDL07 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL07 - Runtime

Objective: This test case executes the NDDL command "Alter Attribute" to alter the ownership of an attribute to another entity. Four alter attribute commands are executed to illustrate how an attribute's ownership can be changed while preserving key migrations if the attribute is key in the new entity, or migrations lost if the attribute is non-key in the new entity.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test case requires the following data files:

CREMOD.DAT	(To create a model)
ALTATTR.DAT	(To manipulate attribute ownership)
DELMOD.DAT	(To drop the model)

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Repeat the above procedure for files ALTATTR.DAT and DELMOD.DAT.

Test Inputs : \$ @NDDL CREMOD

Expected Test Results : The first alter attribute command changes a non-keyed attribute's ownership to a new entity as key.

The second command changes a non-keyed attribute's ownership to a new entity as key and migrates the key back to the old owner.

The third command changes a keyed attribute's ownership to a dependent entity as key, but loses some key migration while preserving the others.

The fourth command alters an attribute's ownership to its independent entity as key and continues preserving key migrations.

Successful Completion Criteria for Test: The execution status will be reported as successful for each command listed in files CREMOD.OUT, ALTATTR.OUT and DELMOD.OUT.

2.44 NDDL08 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL08 - Runtime

Objective: This test case executes the NDDL command "Alter Entity" to demonstrate the capability of altering a tag name, altering a key name and altering key membership.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test case requires the following data files:

CREMOD.DAT (to create the model)
ALTENTK.DAT (to alter the entities)
DELMOD.DAT (to drop the model)

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Repeat the above procedure for ALTENTK.DAT and DELMOD.DAT.

Test Inputs : \$ @NDDL CREMOD

Expected Test Results : The first alter entity command renames a tag (role) name.

The second command renames a key class name.

The third command adds a new key class member, while the fourth drops a key class member from a specified key.

The fourth command changes an alternate key to a primary key and changes the original primary key to an alternate key.

The last command substitutes key members with different attributes of the entity while preserving key migrations of the original key members.

Successful Completion Criteria for Test: The execution status will be reported as successful for each command listed in files CREMOD.OUT, ALTENTK.OUT and DELMOD.OUT.

2.45 NDDL09 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL09 - Runtime

Objective: This test case will execute the following NDDL commands to control the commit/rollback mode, and demonstrate the capability of automatic vs manual commit.

- . Set commit automatic/manual
- . Halt with rollback
- . Commit
- . Rollback

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 3 minutes

Special Resource Considerations : This test case requires the following data files:

SETCOM1.DAT (manual commit)
SETCOM2.DAT (automatic commit)
SETCOM3.DAT (drop the models)

Test Definition

Method of Performing Test: With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Repeat the above procedure for the file SETCOM2.DAT. After the test case is completed, delete the models that were created using SETCOM3.DAT.

Test Inputs : \$ @NDDL SETCOM1

Expected Test Results: The first test case establishes the commit mode as manual. Consequently, the commit and rollback commands are in effect and only one model is created. The second test case establishes the commit mode as automatic. Consequently, the commit and rollback commands are not operative and three models will be created.

Successful Completion Criteria for Test: The execution status and commit status will be reported for each command listed in files SETCOM1.OUT, SETCOM2.OUT and SETCOM3.OUT.

2.46 NDDL10 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL10 - Runtime

Objective: This test case will create a model and attribute in the CDM. The attribute will also be described. This description text will be edited through the text editor in the following test case. A description type will also be created. The test case will execute the following NDDL commands:

```
CREATE MODEL
CREATE ATTRIBUTE
CREATE DESCRIPTION TYPE
DESCRIBE
```

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file DESCRB.DAT which contains the NDDL commands.

Test Definition

Method of Performing Test: With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DESCRB

Expected Test Results: The results of this runtime NDDL test will be creation of a model, an attribute, a line of description text for the attribute and creation of a description type.

The CDM tables affected are:

```
MODEL CLASS
ATTRIBUTE_NAME
ATTRIBUTE_CLASS
DESCRIPTION_TYPE
DESC_TEXT
```

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in DESCRB.OUT.

2.47 NDDL11 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL11 - Runtime

Objective: This test case will create descriptive text for an attribute. This command will be entered from the forms text editor. After describing the object, the model created in NDDL10 will be dropped from the CDM. Dropping the model will drop the attribute and its descriptive text. The test case will execute the following NDDL commands:

```
DESCRIBE
DROP DESCRIPTION TYPE
DROP MODEL
HALT
```

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : Refer to the attached VT100 Function Keypad.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : Following is an example of the Describe command from the FORMS TEXT EDITOR. All bold face lettering is user input.

```
$ @NDDL -I
```

When -I is entered, the NDDL environment (with the IISS UI/VTI) is activated. An NDDL form will be displayed at which time a command may be entered.

```
NEXT COMMAND --
```

```
ALTER MODEL DESC_MODEL;      <ENTER>
```

After the command has been executed a new screen is painted. This screen informs the user a current model (DESC_MODEL) has been established, and no errors were encountered.

```
NEXT COMMAND --
```

DESCRIBE EXPLANATION OF ATTRIBUTE DESC_ATTR; <ENTER>

A new screen appears

<<<<buf_bof>>>>

and this screen also contains the original line of descriptive text. This text may now be edited by adding some more lines using the insert line function key. For example:

THIS IS LINE TWO (2)
THIS IS THE LAST LINE OF DESCRIPTION.

Now to save the changes,

- press <command> key (this takes you to the bottom)
- type SAVE , press <enter> key (this will repaint the screen and tell you the operation is completed.)
- press <quit> key (the original NDDL form will come up with the message that no errors are encountered.)

Now, the attribute has three lines of descriptive text in the CDM.

Clear the screen, and enter another NDDL command.

NEXT COMMAND --

DROP MODEL DESC_MODEL; <enter>

A new screen will appear after the command is executed, with the message that no errors were encountered.

DROP DESCRIPTION TYPE EXPLANATION; <ENTER>

A new screen will appear after the command is executed with the message that no errors were encountered.

NEXT COMMAND --

HALT; <ENTER>

\$

Expected Test Results : After the Describe command is executed, some lines of descriptive text will be created in the CDM for the attribute DESC_ATTR. After the Drop Model command is executed, the model, attribute and descriptive text will be deleted from the CDM. After the Drop description type command, the user defined description type "explanation" is deleted from the CDM.

Successful Completion
Criteria for Test : As above.

pf1 <MODE>	pf2 <HELP>	pf3 <MESSAGE QUEUE>	pf4 <QUIT>
7 <COMMAND>	8	9 <FIRST PAGE>	- <LAST PAGE>
4 <INSERT LINE>	5 <DELETE LINE>	6 <PASTE>	<FILL>
1 <MIDLINE BREAK>	2 <SELECT>	3	<ENTER>
0 <SEARCH NEXT>		<HALT>	

2.48 NDDL12 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL12 - Runtime

Objective: This test case will create a new domain in the CDM along with the domain's data types, values and ranges. The test case will execute the following command:

CREATE DOMAIN

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file CRTDOM.DAT which contains the NDDL commands.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CRTDOM

Expected Test Results: : The results of this runtime NDDL test will be the creation of a new domain in the CDM. The CDM tables affected are:

DOMAIN_CLASS
DOMAIN_RANGE
DOMAIN_VALUE
USER_DEF_DATA_TYPE

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CRTDOM.OUT.

2.49 NDDL13 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL13 - Runtime

Objective: This test case will execute an NDDL command which will change a domain's standard data type and its values and ranges. The test case will execute the following command:

ALTER DOMAIN

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file ALTDOM.DAT which contains the NDDL command. In addition, test case NDDL12 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ALTDOM

Expected Test Results: : The results of this runtime NDDL test will be an altered domain. The CDM tables affected are:

DOMAIN_CLASS
DOMAIN_RANGE
DOMAIN_VALUE
VERIF_MODULE
SOFTWARE_MODULE
MODULE_PARAMETER
USER_DEF_DATA_TYPE

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ALTDOM.OUT.

2.50 NDDL14 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL14 - Runtime

Objective: This test case will execute the NDDL command "COPY DOMAIN" which will generate the NDDL to copy a specified domain. Any associated description text, standard and user defined data types will also be copied as they are not excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPDOM.DAT. In addition, test cases NDDL12 and NDDL13 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPDOM

Expected Test Results : The results of this run time test will be a file (COPDOM.FIL) which contains the NDDL necessary to copy a specified domain. NDDL commands generated are:

```
CREATE DOMAIN TEST CASE STANDARD TYPE TC2
CHARACTER 15
TYPE TC1 SIGNED 6: 2 TYPE TC3
INTEGER 8: 2
VALUE '-7678.79' 'TESTSTRING1'
'TESTSTRING2' ;
```

Successful Completion

Criteria for Test : The execution status will be reported as successful for NDDL command listed in COPDOM.OUT.

2.51 NDDL15 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL15 - Runtime

Objective: This test case will execute an NDDL command to drop a Domain. Values and ranges associated with the domain are deleted and then the domain itself is deleted. The test case will execute the following command:

DROP DOMAIN

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource

Considerations : This test requires the data file DRPDOM.DAT which contains the NDDL command. In addition, test case NDDL12 and NDDL13 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DRPDOM

Expected Test Results: : The results of this runtime NDDL test will be a dropped domain.

The CDM tables affected are:

DOMAIN_CLASS
DOMAIN_RANGE
DOMAIN_VALUE
USER_DEF_DATA_TYPE

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DRPDOM.OUT.

2.52 NDDL16 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL16 - Runtime

Objective: This test will create attributes, entities and relations in the INTEGRATED_MODEL. These conceptual schema objects will be used in the series of test cases NDDL17 through NDDL53 to demonstrate the mapping commands and copy commands (displaying the CDM contents).

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE

CDMP: Distributed Request Supervisor

CDM File/Module Processing
Capabilities

NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test requires the data file PREMAP.DAT which contains the NDDL commands to create the attributes, entities and relations.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL PREMAP

Expected Test Results: : The results of this runtime test will be the creation of seven entities, its attributes, a link relation, a category relation and two keys.

The CDM tables affected are all tables pertaining to Conceptual Schema-- attributes, entities, relations and keys.

Successful Completion Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in PREMAP.OUT.

2.53 NDDL17 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL17 - Runtime

Objective: This test case will execute the NDDL command "CREATE VIEW" which will create a user view in the CDM.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 7 minutes

Special Resource Considerations : This test requires the data file CRTVIEW.DAT which contains the NDDL command. In addition, test case NDDL16 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CRTVIEW

Expected Test Results: : The results of this runtime test will be a newly created view in the CDM.

The CDM tables affected are:

DATA_ITEM
PROJECT_DATA_ITEM
USER_VIEW
VIEW_EC_XREF
VIEW_QUAL_XREF
VIEW_QUALIFY_CRITERIA

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CRTVIEW.OUT.

2.54 NDDL18 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL18 - Runtime

Objective: This test case will execute the NDDL command "COPY VIEW" which will generate the NDDL to copy a specified view. Any associated description text and algorithms will also be copied as they are not excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
COMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource

Considerations : This test case requires the data file COPVIEW.DAT. In addition, test case NDDL17 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPVIEW

Expected Test Results : The results of this runtime test will be an output file (COPVIEW.FIL) which contains the NDDL necessary to copy a view. NDDL commands generated are:

```
CREATE VIEW EMPLOYEE IN DEPARTMENT
DATA ITEM DEPT_NUM TYPE CHARACTER_NAME
EMP_LOCATION TYPE CHARACTER_NAME
AS SELECT DISTINCT
BB.DEPT_NO
AA.E_LOC
FROM
EMP.AA
DEPT.BB
WHERE
BB.DEPT_NO U= AA.DEPT_NO
AND ( ( ( ( BB.DEPT_NO > '400'
OR AA.E_NO = '777' )
AND AA.E_NAME != 'DON' )
OR AA.E_NAME = 'JOHN' )
AND ( (
```

```
(
  BB.DEPT_NO <= '400'
  AND
  AA.E_NO != '777'
)
OR
  AA.E_NAME = 'DON'
)
OR AA.E_NAME != 'JOHN' ) ) ) ;
CREATE VIEW PART_ENTRY
  DATA ITEM PART_ID TYPE CHARACTER_NAME
  PART_INFO TYPE CHARACTER_NAME
  AS SELECT DISTINCT
  AA.PART_ID
  AA.PART_INFO
  FROM
  PART_ENT.AA
  WHERE
  ( AA.PART_ID = 'OH' ) ;
CREATE VIEW MANAGER_VIEW
  DATA ITEM MANAGER_NAME TYPE CHARACTER_NAME
  MANAGER_NUMBER TYPE CHARACTER_NAME
  MANAGER_LOCATION TYPE CHARACTER_NAME
  AS SELECT
  AA.M_NAME
  AA.M_NO
  AA.M_LOC
  FROM
  MANAGER.AA ;
```

Successful Completion

Criteria for Test : The execution status will be reported
as successful for each NDDL command listed in
COPVIEW.OUT.

2.55 NDDL19 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL19 - Runtime

Objective: This test will execute the NDDL command "DROP VIEW" which will delete the user view created in test case NDDL17.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file DRPVIEW.DAT which contains the NDDL command. In addition, test cases NDDL16 and NDDL17 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DRPVIEW

Expected Test Results: : This runtime test case will result in the user view being deleted from the CDM.

The CDM tables affected are:

USER_VIEW
DATA_ITEM
PROJECT_DATA_ITEM
VIEW_EC_XREF
VIEW_QUAL_XREF
VIEW_QUALIFY_CRITERIA

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DRPVIEW.OUT.

2.56 NDDL20 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL20 - Runtime

Objective: This test case will execute the following NDDL commands to define a DB2 DBMS:

DEFINE DBMS

Test cases NDDL20 through NDDL23 define a Relational DBMS's Internal Schema objects which will be used for Conceptual/Internal Schema mappings.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file DEFDDDB.DAT which contains the NDDL commands to define a DBMS.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DEFDDDB

Expected Test Results: : The runtime test case will result in the creation of a relational DBMS - DB2.

The CDM tables affected are:

IISS_DBMS

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DEFDDDB.OUT.

2.57 NDDL21 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL21 - Runtime

Objective: This test case will execute the NDDL command "ALTER DBMS" to associate a particular DBMS with a HOST. The DBMS was created in the CDM in the previous test case.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file ALTDDDB.DAT. In addition, test case NDDL20 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ALTDDDB

Expected Test Results: : This runtime test case will associate the relational DBMS DB2 with the Host VAX.

The CDM tables affected are:

DBMS_ON_HOST

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ALTDDDB.OUT.

2.58 NDDL22 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL22 - Runtime

Objective: This test case will execute the following NDDL commands to define two relational DBMS's databases with tables and columns in the CDM:

DEFINE DATABASE
DEFINE RECORD...WITH FIELDS

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test requires the data file ORCCMD1.DAT which contains the NDDL commands. In addition, test cases NDDL20 and NDDL21 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ORCCMD1

Expected Test Results: : This runtime test case will result in two databases with two records each and some fields being created in the CDM.

The CDM tables affected are:

DATA_BASE
DB_PASSWORD
RECORD_TYPE
DATA_FIELD

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ORCCMD1.OUT.

2.59 NDDL23 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL23 - Runtime

Objective: This test case will execute the NDDL command "ALTER DATABASE" which will assign user defined NULL values for the database and specify where the user's application processes will be stored. The database will have been created in the previous test case.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file ORCCMD2.DAT which contains the NDDL command. In addition, test cases NDDL20 through NDDL22 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ORCCMD2

Expected Test Results: : This runtime test case will result in altering the characteristics of the ORACLE DBMS's database.

The CDM tables affected are:

DATA_BASE

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ORCCMD2.OUT.

2.60 NDDL24 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL24 - Runtime

Objective: This test case will execute the following command to define a HOST to the CDM:

DEFINE HOST

Test cases NDDL24 through NDDL27 define a CODASYL DBMS's Internal Schema object which will be used for Conceptual/Internal Schema Mappings.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file DEFDDDB2.DAT which contains the NDDL commands to define a HOST.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DEFDDDB2

Expected Test Results: : This runtime test case will result in the creation of CYBER in the CDM as a Host computer.

The CDM tables affected are:

IISS_HOST

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DEFDDDB2.OUT.

2.61 NDDL25 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL25 - Runtime

Objective: This test case will execute the NDDL command "ALTER DBMS" to add a host association and then drop a host association. The DBMS and HOST were defined to the CDM in the previous test case.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file ALTDDDB2.DAT which contains the NDDL commands to alter the DBMS HOST associations. In addition, test case NDDL24 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ALTDDDB2

Expected Test Results: : This runtime test case will result in associating a VAX-11 DBMS with the HOST CYBER.

The CDM tables affected are:

DBMS_ON_HOST

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ALTDDDB2.OUT.

2.62 NDDL26 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL26 - Runtime

Objective: This test case will execute the following NDDL commands to define a CODASYL database with associated areas, schemas, records, fields and sets to the CDM:

```
DEFINE DATABASE
DEFINE RECORD with FIELDS
DEFINE SET
```

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file CODCMD1.DAT which contains the NDDL commands to create the CODASYL Internal Schema objects. In addition, the test cases NDDL24 and NDDL25 must have completed successfully.

Test Definition

Method of Performing Test: With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CODCMD1

Expected Test Results: This runtime test case will result in the creation of a VAX-11 database, two records and three sets to the CDM.

The CDM tables affected are:

```
DATA_BASE
DATA_BASE AREA
DB AREA ASSIGNMENT
SCHEMA_NAMES
RECORD_TYPE
DATA_FIELD
RECORD SET
SET_TYPE_MEMBER
```

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in CODCMD1.OUT.

2.63 NDDL27 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL27 - Runtime

Objective: This test case will execute the NDDL command "ALTER FIELD" to change the characteristic of a field in the VAX-11 database.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test case requires the data file CODCMD2.DAT which contains the NDDL command to alter the field. In addition, test cases NDDL24 through NDDL26 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CODCMD2

Expected Test Results: : This runtime test case will result in altering a field which had been defined earlier to the CDM. The field is now defined as key to the record of the VAX-11 database.

The CDM tables affected are:

DATA_FIELD

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CODCMD2.OUT.

2.64 NDDL28 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL28 - Runtime

Objective: This test case will execute the following commands to define a TOTAL DBMS, and a HOST computer IBM:

DEFINE DBMS
DEFINE HOST

Test cases NDDL28 through NDDL31 will define Network type DBMS's Internal Schema objects.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file DEFDDDB3.DAT which contains the NDDL commands to define a DBMS and HOST.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DEFDDDB3

Expected Test Results : This runtime test case will result in the creation of a Network type DBMS (TOTAL) in the CDM. A Host computer, IBM, will also be defined.

The CDM tables affected are:

IISS_DBMS
IISS_HOST

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DEFDDDB3.OUT.

2.65 NDDL29 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL29 - Runtime

Objective: This test case will execute the NDDL command "ALTER HOST" to make a DBMS and HOST association in the CDM. The DBMS and HOST were defined to the CDM in the previous test case.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file ALTDDDB3.DAT which contains the NDDL command. In addition, test case NDDL28 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ALTDDDB3

Expected Test Results: : The results of this runtime test case will be the association of a TOTAL DBMS with the HOST IBM.

The CDM tables affected are:

DBMS_ON_HOST

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ALTDDDB3.OUT.

2.66 NDDL30 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL30 - Runtime

Objective: This test case will execute the following NDDL commands to define a TOTAL database and a complex record in the CDM. The record has subcomponent data fields:

DEFINE DATABASE
DEFINE RECORD

Resource Requirements
Number of terminals : 1
S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file TOTCMD1.DAT which contains the NDDL commands to define the Total Internal Schema objects to the CDM. In addition, test cases NDDL28 and NDDL29 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL TOTCMD1

Expected Test Results: : The results of this runtime test will be the creation of a TOTAL database and a record with a subcomponent field structure.

The CDM tables affected are:

DATA_BASE
DATA_BASE AREA
DB AREA ASSIGNMENT
RECORD TYPE
DATA_FIELD

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in TOTCMD1.OUT.

2.67 NDDL31 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL31 - Runtime

Objective: This test case will execute the following NDDL commands to alter the characteristics of a previously defined record and field in the CDM:

ALTER RECORD
ALTER FIELD

The record and fields of the TOTAL Database being altered were defined in a previous test case. "Alter Record" appends additional subcomponent fields to the existing record structure. "Alter Field" shows the capability of redefining fields, indexing fields and altering a group data type to NULL.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 3 minutes

Special Resource Considerations : This test requires the data file TOTCMD2.DAT which contain the NDDL commands to alter a record and fields characteristics. In addition, test cases NDDL28 through NDDL30 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL TOTCMD2

Expected Test Results: : This runtime test case results in altering a record and its fields characteristics. The record will now contain a more complex structure with levels of subcomponent fields, with fields being redefined, fields that repeat depending on another data field, and group fields having a NULL data type.

The CDM tables affected are:

DATA_FIELD

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in TOTCMD2.OUT.

2.68 NDDL32 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL32 - Runtime

Objective: This test case will execute the following NDDL commands to define a hierarchical DBMS (IMS) and an associated PSB (Program Specification Block) to the CDM.

DEFINE DBMS
DEFINE PSB

Test cases NDDL32 through NDDL35 will define an IMS DBMS's Internal Schema objects.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test case requires the data file DEFDDDB4.DAT which contains the NDDL commands to define a DBMS, make a DBMS and HOST association, and also to define a PSB.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DEFDDDB4

Expected Test Results: : This runtime test case will result in the creation of an IMS DBMS, an IMS to IBM association and a PSB association.

The CDM tables affected are:

IISS_DBMS
DBMS_ON_HOST
IISS_PSB

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DEFDDDB4.OUT.

2.69 NDDL33 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL33 - Runtime

Objective: This test case executes the NDDL command "ALTER PSB" to alter a PSB_HOST association. The PSB and HOST were both defined to the CDM in the previous test case.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource

Considerations : This test requires the data file ALTDDDB4.DAT which contains the NDDL command to change the PSB_HOST association. In addition, the test case NDDL32 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ALTDDDB4

Expected Test Results: : The results of this runtime test case is the new association of an IMS PSB with an IBM HOST computer.

The CDM tables affected are:

IISS_PSB

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ALTDDDB4.OUT.

2.70 NDDL34 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL34 - Runtime

Objective: This test case executes the following NDDL commands to define an IMS PCB and a segment in the CDM. The items/fields in the segment are composed of a subcomponent structure, with some items defined as unique or duplicate key, and one item defined as the COBOL equivalent "FILLER".

DEFINE DATABASE
DEFINE RECORD

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file ISCMDS1.DAT which contains the NDDL commands to define a heirarchical DBMS's Internal Schema. In addition, the test cases NDDL32 through NDDL33 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ISCMDS1

Expected Test Results: : The results of this runtime test is the creation of an IMS PCB and one segment in the PCB with associated items.

The CDM tables affected are:

DATA_BASE
PSB_PCB
RECORD TYPE
DATA_FIELD

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ISCMDS1.OUT.

2.71 NDDL35 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL35 - Runtime

Objective: This test case executes the NDDL command "ALTER RECORD" to alter a previously created segment in the CDM.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test requires the data file ISCMDS2.DAT which contains the NDDL commands to alter a record and drop a field. In addition, the test cases NDDL32 through NDDL34 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ISCMDS2

Expected Test Results: : This runtime test case results in altering an IMS segment, dropping a field and its subcomponent fields.

The CDM tables affected are:

DATA_FIELD

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in ISCMDS2.OUT.

2.73 NDDL37 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL37 - Runtime

Objective: This test case executes the following NDDL commands:

"ALTER PARTITION": To add and drop a fragment record of the partitioned entity.

"ALTER UNION": To add and drop entities of the record union.

"ALTER MODULE": To add more parameters to the software module definition.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 4 minutes

Special Resource Considerations: This test requires the data file ORCMAP2.DAT which contains the NDDL commands to alter the previously defined Partition, Union and Module. In addition, test case NDDL36 must have completed successfully.

Test Definition

Method of Performing Test: With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ORCMAP2

Expected Test Results: This runtime test case results in altering the definitions of the Horizontal Partition, Record Union and Software Module.

The CDM tables affected are:

HORIZONTAL_PART
ECRTUD
SOFTWARE_MODULE
MODULE_PARAMETER

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in ORCMAP2.OUT.

2.74 NDDL38 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL38 - Runtime

Objective: This test case creates Conceptual to Internal Schema mappings in the CDM. The following NDDL commands are executed:

"CREATE MAP": To create Attribute Use Class (Tag) to data field mappings for different preferences.

"DEFINE ALGORITHM": To define the use of a Software Module as a complex mapping algorithm.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 4 minutes

Special Resource Considerations : This test requires the data file ORCMAP3.DAT which contains the NDDL commands to define the CS/IS Mappings to the CDM. In addition, test cases NDDL36 and NDDL37 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ORCMAP3

Expected Test Results: : The results of this runtime test case is the creation of a CS/IS Mapping for:

- One tag (that belongs in a horizontally partitioned entity) to two data fields for Preference 1.

- A tag mapped to two fields for Preference 1 and 2.

- A complex mapping algorithm used for an update application.

The CDM tables affected are:

AUC_IS_MAPPING
PROJECT_DATA_FIELD
COMPLEX_MAPPING_PARM

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in ORCMAP3.OUT.

2.75 NDDL39 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL39 - Runtime

Objective: This test case executes the NDDL command "ALTER MAP" to alter a previously defined mapping. The first command drops a field mapping and the second Alter Map command switches mapping preferences.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file ORCMAP4.DAT which contains the NDDL Alter Map command. In addition, test cases NDDL36 through NDDL38 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ORCMAP4

Expected Test Results: : The results of this runtime test are an altered mapping definition in the CDM.

The affected CDM tables are:

AUC IS MAPPING
PROJECT_DATA_FIELD

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in ORCMAP4.OUT.

2.76 NDDL40 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL40 - Runtime

Objective: This test case executes the NDDL command "CREATE MAP" to define a Conceptual to Internal Schema Mapping for a CODASYL DBMS. The first command maps an Attribute Use Class (Tag) to Record Sets for a stated preference. The second command maps a relation class to a record set.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file CODMAP1.DAT which contains the NDDL command to create a tag to set mapping. In addition, test cases NDDL16 and NDDL24 through NDDL27 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CODMAP1

Expected Test Results: : The results of this runtime test case are the creation of a CS/IS mapping (Tag to record sets) for Preference 1 and a relation to record set mapping.

The CDM tables affected are:

AUC_IS_MAPPING
AUC_ST_MAPPING
RC_BASED_REC_SET

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CODMAP1.OUT.

2.77 NDDL41 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL41 - Runtime

Objective: This test case executes the NDDL command "ALTER MAP" to alter a stated preference mapping between Attribute Use Class (Tag) and Record Sets. This command adds a new set mapping, alters a set value, and finally drops a previously defined set mapping.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file CODMAP2.DAT which contains the NDDL command to alter a mapping definition. In addition, test case NDDL40 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CODMAP2

Expected Test Results: : The result of this runtime test case is an altered mapping definition.

The CDM tables affected are:

AUC_ST_MAPPING

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CODMAP2.OUT.

2.78 NDDL42 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL42 - Runtime

Objective: This test case will execute the NDDL command "COPY MODULE" which will generate the NDDL necessary to copy a specified software module.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPMODU.DAT. In addition, test cases NDDL36 and NDDL37 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPMODU

Expected Test Results: : The results of this runtime test will be a file (COPMODU.FIL) which contains the NDDL necessary to copy a software module. NDDL commands generated are:

```
DEFINE MODULE CONVALG IN COBOL
PARAMETERS
IN_PARM1 TYPE CHARACTER_NAME
IN_PARM2 TYPE CHARACTER_NAME
OUT_PARM1 TYPE CHARACTER_NAME
OUT_PARM2 TYPE RET_STATUS ;
```

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in COPMODU.OUT.

2.79 NDDL43 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL43 - Runtime

Objective: This test case will execute the NDDL command "COPY DBMS INCLUDE DATABASE EXCEPT DESCRIPTION" which will generate the NDDL to copy a specified DBMS. The description text associated with the DBMS will not be copied as descriptions are excepted. All databases associated with the IMS DBMS with their records and fields are copied as the database option has been included.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPDBMS.DAT. In addition, test cases NDDL32 through NDDL35 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPDBMS

Expected Test Results: : The results of this runtime test will be a file (COPDBMS.FIL) which contains the NDDL necessary to copy a DBMS and its associated databases. NDDL commands generated are:

```
DEFINE DBMS IMS MODEL H ON HOST IBM CYBER ;
DEFINE IMS PCB NAMED IMS_DB ON HOST IBM
  POSITION 1 IN PSB IMS_PSB1 FEEDBACK LENGTH 30
  STORES CHARACTER NULL AS ZEROS
  STORES INTEGER NULL AS ZEROS NTM DIRECTORY 'GR' ;
ALTER DATABASE IMS_DB ;
DEFINE RECORD STUDENT WITH FIELDS 1 STU_NAME DATA TYPE
  CHARACTER_NAME KNOWN UNIQUE KEY ;
```

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in COPDBMS.OUT.

2.80 NDDL44 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL44 - Runtime

Objective: This test case will execute the NDDL command "COPY HOST ALL" which will generate the NDDL necessary to copy all the host computers defined to the CDM. Any associated PSB's and description text will also be copied as they are not excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPHOST.DAT. In addition, test cases NDDL20, NDDL21, NDDL24, NDDL28, NDDL29, NDDL32 and NDDL33 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPHOST

Expected Test Results: : The results of this runtime test will be a file (COPHOST.FIL) which contains the NDDL necessary to copy all hosts with associated descriptive test and PSBs. NDDL commands generated are:

```
DEFINE HOST VAX ;  
DEFINE HOST CYBER ;  
DEFINE HOST IBM ;  
DEFINE PSB IMS_PSB1 ON HOST IBM ;  
DEFINE PSB IMS_PSB2 ON HOST IBM ;
```

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in OPHOST.OUT.

2.81 NDDL45 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL45 - Runtime

Objective: This test case will execute the command "COPY DATABASE EXCEPT DESCRIPTION" which will generate the NDDL to copy a specified ORACLE database along with its records and fields. Any associated descriptive text will not be copied as descriptions have been excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPDB.DAT. In addition, test cases NDDL20 through NDDL23 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPDB

Expected Test Results: : The results of this runtime test will be a file (COPDB.FIL) which contains the NDDL necessary to copy a specified database with its associated records and fields. NDDL commands generated are:

```
DEFINE ORACLE DATABASE NAMED ORC_DB ON HOST
                                VAX WITH PASSWORD CDC
                                STORES CHARACTER NULL AS SPACES
STORES INTEGER NULL AS ZEROS NTM
                                DIRECTORY 'GR' ;
ALTER DATABASE ORC_DB ;
DEFINE RECORD DEPT WITH FIELDS
    1 DEPT_NO DATA TYPE
      CHARACTER_NAME KNOWN
    1 DEPT_NAME DATA TYPE
      CHARACTER_NAME KNOWN
    1 DEPT_LOC DATA TYPE
      CHARACTER_NAME KNOWN
;
```

UTP620341000
30 September 1990

```
DEFINE RECORD PART WITH FIELDS
  1 PART_NO DATA TYPE
    CHARACTER_NAME KNOWN
  1 PART_DESC DATA TYPE
    CHARACTER_NAME KNOWN
  1 PART_TYPE DATA TYPE
    CHARACTER_NAME KNOWN
;
```

Successful Completion

Criteria for Test: : The execution status will be reported
as successful for each NDDL command listed in
COPDB.OUT.

2.82 NDDL46 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL46 - Runtime

Objective: This test case will execute the NDDL command "COPY RECORD ALL OF DATABASE" which will generate the NDDL to copy all records and fields of a specified database. Descriptive text for the database, records and fields are also generated as they are not excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPREC.DAT. In addition, test cases NDDL28 through NDDL31 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPREC

Expected Test Results: : The results of this runtime test will be a file (COPREC.FIL) which contains the NDDL necessary to copy a database with all its records, fields and associated descriptive text. NDDL commands generated are:

```
ALTER DATABASE TOT_DB ;
DEFINE RECORD EMPL WITH FIELDS
  1 EMPLNAME DATA TYPE NUMERIC ID KNOWN
  1 EMPLINFO OCCURS 10 DEPENDING ON EMPLNAME
    INDEXED BY EMPLINDEX KNOWN
  2 EMPLNO DATA TYPE CHARACTER NAME KNOWN
  2 EMPLADDR REDEFINES EMPLNO KNOWN
  3 EMPLCITY DATA TYPE CHARACTER_NAME KNOWN
  3 EMPLSTAT DATA TYPE CHARACTER_NAME KNOWN ;
```

Successful Completion Criteria for Test: The execution status will be reported as succesful for each NDDL command listed in COPREC.OUT.

2.83 NDDL47 - Runtime

Subsystem: : CDM Release: 3.0

Test Name and Number : NDDL47 - Runtime

Objective: This test case will execute the NDDL command "COPY SET ALL OF DATABASE EXCEPT DESCRIPTION" which will generate the NDDL to copy all sets of a specified CODASYL database. Any associated descriptive text will be copied as descriptions have not been excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPSET.DAT. In addition, test cases NDDL24 through NDDL27 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPSET

Expected Test Results: : The results of this runtime test will be a file (COPSET.FIL) which contains the NDDL necessary to copy all sets of a database. NDDL commands generated are:

```
ALTER DATABASE COD DB ;  
DEFINE SET DOMESTIC RELATING SHIP_INFO TO  
SHIP_LOC REQUIRED ;  
DEFINE SET EXPORT RELATING SHIP_INFO TO  
SHIP_LOC OPTIONAL ;  
DEFINE SET IMPORT RELATING SHIP_INFO TO  
SHIP_LOC OPTIONAL ;  
DEFINE SET SHIP_EMPL RELATING SHIP_LOC TO  
EMPL REQUIRED ;
```

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in COPSET.OUT.

2.84 NDDL48 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL48 - Runtime

Objective: This test case will execute the NDDL commands "COPY MAP FOR ENTITY", "COPY MAP FOR RELATION", "COPY MAP FOR RECORD" and "COPY MAP FOR SET", each of which will generate the NDDL commands to copy conceptual schema to internal schema mapping definitions. NDDL will be generated to copy complex mapping algorithms, record unions and horizontal partitions as they are not excepted. Maps are copied via conceptual schema entities and relations and also via internal schema records and sets.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPMAP.DAT. In addition, test cases NDDL16 thru NDDL41 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPMAP

Expected Test Results: : The results of this runtime test will be 4 files (COPMAPE.FIL, COPMAPT.FIL, COPMAPR.FIL, COPMAPS.FIL) which contain the NDDL necessary to copy CS/IS mappings. NDDL generated in the following files are:
COPMAPE.FIL

```
CREATE MAP PART_ENT TO RECORD ORC_DB.PART
ORC_DB2.PARTS
ALLOW RETRIEVAL ALLOW UPDATE ;
CREATE MAP SHIP_DEPT TO RECORD
COD_DB.SHIP INFO
ALLOW RETRIEVAL DISALLOW UPDATE ;
CREATE PARTITION 1 OF ENTITY PART_ENT
TO RECORD
ORC_DB.PART
ORC_DB2.PARTS
;
```

```
CREATE MAP PART_ENT.  
  PART_INFO ACTIVE ORIGINAL_SOURCE  
  FOR PREFERENCE 1  
  TO FIELD ORC_DB.  
  PART.  
  PART_DESC  
  ;  
CREATE MAP SHIP_DEPT.  
  SHIP_TYPE ACTIVE ORIGINAL_SOURCE  
  FOR PREFERENCE 1  
  TO SET COD_DB.EXPORT VALUE 'EXPORT'  
  COD_DB.IMPORT VALUE 'IMPORT'  
  ;  
COPMAPT.FIL  
  
CREATE MAP MANAGER TO RECORD ORC_DB.DEPT  
  DISALLOW RETRIEVAL DISALLOW_UPDATE ;  
CREATE MAP EMP TO RECORD ORC_DB.DEPT  
  DISALLOW RETRIEVAL DISALLOW_UPDATE ;  
CREATE MAP DEPT TO RECORD ORC_DB.DEPT  
  ORC_DB2.PARTS ORC_DB2.CAR PARTS  
  ALLOW RETRIEVAL DISALLOW_UPDATE ;  
CREATE MAP DEPT.  
  DEPT_NO ACTIVE ORIGINAL_SOURCE  
  FOR PREFERENCE 2  
  TO FIELD ORC_DB.  
  DEPT.  
  DEPT_NO  
  ;  
DEFINE ALGORITHM CONVALG 01 FOR UPDATE  
  FOR PREFERENCE 01 USING PARAMETERS  
  IN_PARM1 FROM ATTRIBUTE MANAGER.M_LOC  
  
  IN_PARM2 CONSTANT '2.5'  
  OUT_PARM1 TO DATAFIELD  
  ORC_DB.DEPT.DEPT_LOC  
  STATUS  
  ;  
CREATE UNION OF RECORD ORC_DB.DEPT  
  TO ENTITY MANAGER WHEN DEPT_NO < '10'  
  EMP WHEN DEPT_NO >= '10'  
  ;  
COPMAPR.FIL  
  
CREATE MAP DEPT HAS EMP  
  TO SET COD_DB.SHIP_EMPL.EMPL  
  ;  
COPMAPS.FIL  
  
CREATE MAP DEPT HAS EMP  
  TO SET COD_DB.SHIP_EMPL.EMPL  
  ;
```

Successful Completion

Criteria for Test: : The execution status will be reported
as successful for each NDDL command listed in COPMAP.OUT.

2.85 NDDL49 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL49 - Runtime

Objective: This test case executes the NDDL command "DROP MAP" to delete the mapping definition between Attribute Use Class and Record Sets for a stated preference and relation class to set mappings.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file CODMAP3.DAT which contains the NDDL command. In addition, test cases NDDL40 and NDDL41 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CODMAP3

Expected Test Results: : The runtime test case results in dropping a CS/IS mapping for a stated preference from the CDM.

The CDM tables affected are:

AUC_IS_MAPPING
AUC_ST_MAPPING

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CODMAP3.OUT.

2.86 NDDL50 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL50 - Runtime

Objective: This test case executes the following NDDL commands:

"DROP ALGORITHM": To delete the complex mapping algorithm.

"DROP MAP": To delete all mappings of all tags of a specified entity.

"DROP MODULE": To delete the software module itself.

"DROP UNION": To delete the record union of the specified entities.

"DROP PARTITION": To delete the partitioned records of the specified entity.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test requires the data file ORCMAP5.DAT which contains the NDDL commands to delete all mappings created in test cases NDDL36 through NDDL41.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL ORCMAP5

Expected Test Results: : This test case deletes all CS/IS mappings created in the CDM.

The CDM tables affected are:

AUC_IS_MAPPING
PROJECT_DATA_FIELD
COMPLEX_MAPPING_PARM
ECRTUD
HORIZONTAL_PART
SOFTWARE_MODULE
MODULE_PARAMETER
EC_RT_MAPPING

UTP620341000
30 September 1990

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in ORCMAP5.OUT.

2.87 NDDL51 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL51 - Runtime

Objective: This test case deletes the entities, attributes, keys and relations created in test case NDDL16. It is the last in the series of test cases NDDL16 through NDDL51. The test case executes the following NDDL commands.

DROP ENTITY
DROP ATTRIBUTE

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
COMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 4 minutes

Special Resource Considerations : This test requires the data file PSTMAP.DAT which contains the NDDL commands. All tests cases in the series NDDL16 through NDDL50 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL PSTMAP

Expected Test Results: : The result of this runtime test case is the deletion of all attributes, entities, keys and relations (created in test case NDDL16) from the CDM.

The CDM tables affected are all conceptual schema objects--entities, attributes, key and relation classes.

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in PSTMAP.OUT.

2.88 NDDL52 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL52 - Runtime

Objective: This test case deletes all Internal Schema definitions (Database, Records, Sets and Fields) that were created, modified and referenced in test cases NDDL20 through NDDL50. The NDDL commands executed are:

DROP DATABASE
DROP RECORD
DROP FIELD
DROP SET

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 10 minutes

Special Resource Considerations : This test requires the data file DRPFERSD.DAT which contains the NDDL commands to delete Internal Schema object definitions. In addition, test cases NDDL20 through NDDL51 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DRPFERSD

Expected Test Results: : This runtime test case deletes fields, records, sets and databases that are specified.

The CDM tables accessed are all Internal Schema tables pertaining to Records, Fields, Sets, Areas, Schemas, and Database.

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DRPFERSD.OUT.

2.89 NDDL53 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL53 - Runtime

Objective: This test case deletes all PSBs, DBMSs and HOSTs created, modified and referenced in test cases NDDL20 through NDDL35. The NDDL commands executed are:

```
DROP PSB
ALTER HOST...DROP DBMS
DROP DBMS
DROP HOST
```

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test requires the data file DRPDDB.DAT which contains the NDDL commands to delete the DBMS, HOST and PSB definitions in the CDM. In addition, test cases NDDL20 through NDDL35 and NDDL52 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DRPDDB

Expected Test Results: : This runtime test case deletes the earlier defined DBMSs, HOSTs and PSBs from the CDM.

The CDM tables affected are:

```
DBMS_ON_HOST
IISS_DBMS
IISS_HOST
IISS_PSB
```

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in DRPDDB.OUT.

2.90 NDDL54 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL54 - Runtime

Objective: This test will create five (5) conceptual schema models in the CDM to be used in test cases NDDL55 through NDDL66.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 15 minutes

Special Resource Considerations : This test case requires the following five data files which contain NDDL commands to create the models:

MODEL1.DAT
MODEL2.DAT
MODEL3.DAT
MODEL4.DAT
MODEL5.DAT

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below. Repeat the procedure for files MODEL2.DAT, MODEL3.DAT, MODEL4.DAT, and MODEL5.DAT.

Test Inputs : \$ @NDDL MODEL1

Expected Test Results: : After completion of each procedure a newly created model will exist in the CDM. The CDM tables affected are: All conceptual schema tables pertaining to entity, attribute, relations and keys as well as all tables pertaining to keywords, alias and descriptions.

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in MODEL1.OUT, MODEL2.OUT, MODEL3.OUT, MODEL4.OUT and MODEL5.OUT.

2.91 NDDL55 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL55 - Runtime

Objective: This test case will execute the NDDL command "COPY DESCRIPTION" which will copy a description of an object type from an existing model to the same object type in another model.
Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test case requires 2 data files, CPYDES1.DAT which creates the target model and CPYDES.DAT which contains the NDDL command. In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below. Repeat the procedure for file CPYDES.DAT.

Test Inputs : \$ @NDDL CPYDES1

Expected Test Results: : The results of this run time test will be a newly created description for relation NEWENT_B
ALSO_USES NEWENT_C.

The CDM tables affected are:

DESC_TEXT

Successful Completion

Criteria for Test: : The execution status will be reported as successful for each NDDL command listed in CYPDES1.OUT and CYPDES.OUT.

2.92 NDDL56 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL56 - Runtime

Objective: This test case will execute the NDDL command "COPY ATTRIBUTE" to generate the NDDL commands on a specified file. These NDDL commands will copy an attribute from an existing model along with any associated keywords, aliases and descriptions to a target model.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 1 minute

Special Resource Considerations : This test case requires the data file COPATT.DAT which contains the NDDL command "COPY ATTRIBUTE". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPATT

Expected Test Results: : The following NDDL commands to create the attribute will be generated on file COPATT.FIL

```
ALTER MODEL GENERAL PURPOSE;  
CREATE ATTRIBUTE NEWATT_A DOMAIN  
    CHARACTER NAME KEYWORD ATTA_BOY;  
DESCRIBE DEFINITION OF ATTRIBUTE_NEWATT_A  
    "THIS IS AN ALIAS OF ATT_A IN MODEL AUGIE_MOD"  
;  
CREATE ALIAS ATTRIBUTE NEWATT_A IS ATTA_ALIAS;
```

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in COPATT.OUT.

2.93 NDDL57 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL57 - Runtime

Objective: This test case will execute the NDDL command "CHECK MODEL" which will verify if a model conforms to all specified modeling rules.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file CHKMOD.DAT which contains the NDDL command "CHECK MODEL". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CHKMOD

Expected Test Results: : Following are the error messages which result for this model. The messages will be found in CHKMOD.FIL

```
MODEL BEING CHECKED: CRUMMY_MOD 10-25-89 13:12:51 ;
A NON SPECIFIC RELATIONSHIP EXISTS BETWEEN CRUM1
NEEDS CRUM2
ATTRIBUTE/TAG AA FOR ENTITY CRUM1 HAS NO DOMAIN
C2K1 OF CRUM2 IS A DUPLICATE KEY OR SUBSET
WARNING: ENTITY CRUM2 HAS NO OWNED ATTRIBUTES
AN INCOMPLETE RELATION EXISTS BETWEEN CRUM3
WANTS TO OWN CRUM1
ENTITY CRUM3 HAS NO KEY CLASS
ATTRIBUTE/TAG XX FOR ENTITY CRUM3 HAS NO DOMAIN
ATTRIBUTE/TAG YY FOR ENTITY CRUM3 HAS NO DOMAIN
A CATEGORY RELATION EXISTS THAT HAS LESS TWO
MEMBERS FOR GENERIC ENTITY CRUM4 AND CATEGORY
RELATION DUMCAT
ATTRIBUTE/TAG CC FOR ENTITY CRUM4 HAS NO DOMAIN
WARNING: ENTITY CRUM5 HAS NO OWNED
```

ATTRIBUTES

MODEL CRUMMY_MOD HAS NO TOP ENTITY
MODEL CRUMMY_MOD HAS NO BOTTOM ENTITY

Successful Completion

Criteria for Test: : This test case will execute
successfully if the above error messages are returned.

2.94 NDDL58 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL58 - Runtime

Objective: This test case will execute the NDDL command "COMPARE MODEL" to compare two models. The first test case compares the models based on alias or primary names. The second test case compares the models based solely on primary entity and attribute names. All matching attributes, entities and keywords found in the two models are reported to the user.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test case requires the data file CMPMOD.DAT which contains the NDDL command "COMPARE MODEL". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CMPMOD

Expected Test Results: : Following are the messages which are the result of the comparison between the two models. The messages will be found in file CMPMOD.FIL

```
ENTITY OR ALIAS ENT_A EXISTS IN BOTH MODELS
ENTITY OR ALIAS ENT_B EXISTS IN BOTH MODELS
ENTITY OR ALIAS ENT_C EXISTS IN BOTH MODELS
ENTITY OR ALIAS EMPLOYEE EXISTS IN BOTH MODELS
ATTRIBUTE OR ALIAS AA EXISTS IN BOTH MODELS
ATTRIBUTE OR ALIAS BB EXISTS IN BOTH MODELS
ATTRIBUTE OR ALIAS CC EXISTS IN BOTH MODELS
ATTRIBUTE OR ALIAS PAY TYPE EXISTS IN BOTH MODELS
ENTITY ENT_C AND ENTITY ENT_B HAVE MATCHING KEYWORD K3
ATTRIBUTE CC AND ATTRIBUTE AA DD HAVE
MATCHING KEYWORD CC KEYWORD
ATTRIBUTE AA AND ATTRIBUTE CC HAVE MATCHING KEYWORD
AA KEYWORD
ATTRIBUTE ATT_A AND ATTRIBUTE BB HAVE MATCHING KEYWORD
```

ATTA_BOY
LINK RELATION ENT_B USES ENT_C AND LINK
RELATION ENT_X USES ENT_B HAVE MATCHING KEYWORD
USES_KEYWORD
RELATION CLASS ENT1_OWNS ENT_B EXISTS IN BOTH MODELS
COMPLETE CATEGORY RELATION EMPLOYEE_PAYROLL EXISTS IN
BOTH MODELS
CATEGORY RELATION EMPLOY_PAYROLL HAS THE SAME NUMBER OF
MEMBERS IN BOTH MODELS

PRIMARY ENTITY ENT_B EXISTS IN BOTH MODELS
PRIMARY ENTITY ENT1 EXISTS IN BOTH MODELS
PRIMARY ENTITY EMPLOYEE EXISTS IN BOTH MODELS
PRIMARY ATTRIBUTE BB EXISTS IN BOTH MODELS
PRIMARY ATTRIBUTE CC EXISTS IN BOTH MODELS
PRIMARY ATTRIBUTE PAY TYPE EXISTS IN BOTH MODELS
ENTITY ENT_C AND ENTITY ENT_B HAVE MATCHING KEYWORD K3
ATTRIBUTE CC AND ATTRIBUTE AA_DD HAVE MATCHING KEYWORD
CC_KEYWORD
ATTRIBUTE AA AND ATTRIBUTE CC HAVE MATCHING
KEYWORD AA_KEYWORD
ATTRIBUTE ATT_A AND ATTRIBUTE BB HAVE MATCHING
KEYWORD ATTA_BOY
LINK RELATION ENT_B USES ENT_C AND LINK RELATION ENT_X
USES ENT_B HAVE MATCHING KEYWORD USES_KEYWORD
RELATION CLASS ENT1_OWNS ENT_B EXISTS IN BOTH MODELS
RELATION CLASS ENT1_OWNS ENT_B EXISTS IN BOTH MODELS
RELATION CLASS ENT1_OWNS ENT_B EXISTS IN BOTH MODELS
COMPLETE CATEGORY RELATION EMPLOYEE_PAYROLL EXISTS IN
BOTH MODELS ATTRIBUTE PAY TYPE IN BOTH MODELS
CATEGORY RELATION EMPLOYEE_PAYROLL HAS THE SAME
DISCRIMINATING ATTRIBUTE PAY TYPE IN BOTH MODELS
CATEGORY RELATION EMPLOYEE_PAYROLL HAS THE SAME NUMBER
OF MEMBERS IN BOTH MODELS
COMPLETE CATEGORY RELATION EMPLOYEE_PAYROLL EXISTS IN
BOTH MODELS
CATEGORY RELATION EMPLOYEE_PAYROLL HAS THE SAME
DISCRIMINATING ATTRIBUTE PAY TYPE IN BOTH MODELS
CATEGORY RELATION EMPLOYEE_PAYROLL HAS THE SAME NUMBER
OF MEMBERS IN BOTH MODELS

Successful Completion Criteria for Test: The execution status will
be reported as successful for each NDDL command listed in
CMPMOD.OUT.

2.95 NDDL59 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL59 - Runtime

Objective: This test case will execute the NDDL command "COPY ENTITY WITH RELATION" to generate NDDL commands on a specific file. All relations in which the entity being copied is the dependent entity are generated, providing the independent entities in the relation exist in the target model. All relations in which the entity being copied is the independent entity are generated, providing the dependent entities in the relation exist in the target model.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 3 minutes

Special Resource Considerations : This test case requires two data files, COPNTR1.DAT which creates the target model and COPENTR.DAT which contains the NDDL command "COPY ENTITY...WITH RELATION". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below. Repeat the procedure for file COPENTR.DAT.

Test Inputs : \$ @NDDL COPNTR1

Expected Test Results: : Each NDDL command and its execution status will be reported in the files COPNTR1.OUT and COPENTR.OUT.

The following NDDL commands to copy the entity with relation are generated on file COPENTR.FIL

```
ALTER MODEL COPYREL ;
CREATE ENTITY ENT_B;
CREATE ATTRIBUTE ATT_B DOMAIN NUMERIC_ID ;
ALTER ENTITY ENT_B OWNED ATTRIBUTE ATT_B ;
CREATE RELATION 1 ENT_A HAS 0 : MANY ENT_B MIGRATES EAK
SET AA = AA ;
CREATE RELATION 1 ENT1 OWNS 0 : MANY ENT_B MIGRATES E1K
SET EB_BB = BB ;
ALTER ENTITY ENT_B ADD
PRIMARY KEY EBK = AA EB_BB ;
/*THIS MAY FAIL, IF KEY NAME ALREADY EXISTS IN THE
```



```
NEW ENTITY */ ;  
CREATE RELATION 1 ENT_B USES 0 : MANY ENT_C MIGRATES EBK  
SET EC_AA = AA EC_BB = EB_BB ;
```

Successful Completion

Criteria for Test: : The execution status will be reported
as successful for each NDDL command listed in COPNTR1.OUT and
COPENTR.OUT.

2.96 NDDL60 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL60 - Runtime

Objective: This test case will execute the NDDL command "COPY ENTITY WITH STRUCTURE" to generate NDDL commands on a specified file. These NDDL commands will copy the entity, the tree structure dependent on the entity, and all associated attributes, keys and relations. Keywords, aliases and descriptions have been excepted, and are not copied to the target model.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 3 minutes

Special Resource Considerations : This test case requires the data file COPENPTS.DAT which contains the NDDL command "COPY ENTITY WITH STRUCTURE". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPENPTS

Expected Test Results: : Each NDDL command and its execution status will be reported in the file COPENPTS.OUT.

The following NDDL commands to copy the entity with structure are generated on file COPENPTS.FIL.

```
ALTER MODEL GENERALPURPOSE ;
CREATE ENTITY ENT8 ;
CREATE ATTRIBUTE ATT_8 DOMAIN UNDEFINED ;
ALTER ENTITY ENT8 ADD OWNED ATTRIBUTE ATT_8 ;
/* CREATE ENTITY COMMANDS COULD BE DUPLICATED
IF AN ENTITY IS IN BOTH A CATEGORY AND LINK RELATION */
CREATE ENTITY ENT_B ;
CREATE ATTRIBUTE ATT_B DOMAIN NUMERIC_ID ;
ALTER ENTITY ENT_B ADD OWNED ATTRIBUTE ATT_B ;
/* CREATE ENTITY COMMANDS COULD BE DUPLICATED
IF AN ENTITY IS IN BOTH A CATEGORY AND LINK RELATION */
CREATE ENTITY ENT_C ;
```

```
CREATE ATTRIBUTE CC DOMAIN CHARACTER NAME ;
CREATE ATTRIBUTE ATT_C DOMAIN NUMERIC ID ;
ALTER ENTITY ENT_C ADD OWNED ATTRIBUTE CC ATT_C ;
/* CREATE ENTITY COMMANDS COULD BE DUPLICATED
   IF AN ENTITY IS IN BOTH A CATEGORY AND LINK RELATION */
CREATE ENTITY ENT6 ;
CREATE RELATION 1 ENT8 IS_PART_OF 0 : MANY ENT_B ;
/* ALTER ENTITY, ADD KEYS COULD BE REDUNDANT
   IF ENTITY WAS A CATEGORY MEMBER */
CREATE RELATION 1 ENT8 IS_PART_OR 0 : MANY ENT_B ;
ALTER ENTITY ENT_B ADD
    PRIMARY KEY EBK = AA EB_BB ;
CREATE RELATION 1 ENT_B USES 0 UES 0 : MANY ENT_C MIGRATES
    EBK SET
    EC_AA = AA EC_BB = EB_BB ;
ALTER ENTITY ENT_C ADD
    PRIMARY KEY ECK = CC EC_AA ;
CREATE RELATION 1 ENT_B IDENTIFIES 0 : MANY ENT6 MIGRATES
    EBK SET
    AA = AA E6_BB = EB_BB ;
ALTER ENTITY ENT6 ADD
    PRIMARY KEY E6K = E6_BB ;
```

Successful Completion

Criteria for Test: : The execution status will be reported
as successful for each NDDL command listed in
COPENTS.OUT.

2.97 NDDL61 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL61 - Runtime

Objective: This test case will execute the NDDL command "COPY ENTITY" four times which will generate the NDDL necessary to copy specified entities with different combinations of exceptions.

The first command excepts attributes, hence just the Create Entity command is generated. Since the specified entity does not have keywords, aliases or description text, these commands are not generated.

The second command excepts non-keyed attributes when copying a specified entity. This copy generates commands to create the entity and owned keyed attributes with associated keywords, aliases and description text, if any. The key class of the specified entity is also copied.

The third command excepts description, alias keywords and non-keys. This generates the same commands as the above without the descriptions, aliases and keywords.

Finally, the fourth command copies a specified entity with its subordinate structure, but specifies two dependent entities to be excluded.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test requires the data file COPENT1.DAT. In addition, test case NDDL54 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPENT1

Expected Test Results: : The results of this runtime test will be 4 files which contain the NDDL necessary to copy specified entities with different exceptions. NDDL generated will be:

COPENT1.FIL
ALTER MODEL GENERALPURPOSE ;
CREATE ENTITY ENT_A ;

COPENT2.FIL
ALTER MODEL GENERALPURPOSE ;
CREATE ENTITY ENT_A ;
CREATE ATTRIBUTE AA DOMAIN CHARACTER NAME ;
ALTER ATTRIBUTE AA ADD KEYWORD AA KEYWORD ;
ALTER ENTITY ENT_A ADD OWNED ATTRIBUTE AA ;
ALTER ENTITY ENT_A ADD PRIMARY KEY EAK = AA ;

COPENT3.FIL
ALTER MODEL GENERALPURPOSE ;
CREATE ENTITY ENT1_NEW ;
CREATE ATTRIBUTE BB DOMAIN UNDEFINED ;
ALTER ENTITY ENT1_NEW ADD OWNED ATTRIBUTE BB ;
ALTER ENTITY ENT1_NEW ADD PRIMARY KEY E1K = BB ;

COPENT4.FIL
ALTER MODEL GENERALPURPOSE ;
CREATE ENTITY ENT1 ;
CREATE ATTRIBUTE BB DOMAIN UNDEFINED ;
CREATE ATTRIBUTE DD DOMAIN NUMERIC ID ;
ALTER ENTITY ENT1 ADD OWNED ATTRIBUTE BB DD ;
ALTER ENTITY ENT1 ADD
PRIMARY KEY E1K = BB ;
/* THIS MAY FAIL, IF KEY NAME ALREADY EXISTS IN THE NEW
ENTITY */ ;
/* CREATE ENTITY COMMANDS COULD BE DUPLICATED IF AN
ENTITY IS IN BOTH A CATEGORY AND LINK RELATION */
CREATE ENTITY ENT_B ;
ALTER ENTITY ENT_B ADD OWNED ATTRIBUTE ATT_B ;
CREATE ENTITY ENT8 ;
CREATE ATTRIBUTE ATT_8 DOMAIN UNDEFINED ;
ALTER ENTITY ENT8 ADD OWNED ATTRIBUTE ATT_8 ;
/*CREATE ENTITY COMMANDS COULD BE DUPLICATED
IF AN ENTITY IS IN BOTH A CATEGORY AND LINK RELATION */
CREATE ENTITY ENT_B
CREATE ATTRIBUTE ATT_B DOMAIN NUMERIC ID ;
ALTER ENTITY ENT_B ADD OWNED ATTRIBUTE ATT_B ;
/* ALTER ENTITY, ADD KEYS COULD BE REDUNDANT
IF ENTITY WAS A CATEGORY MEMBER */
CREATE RELATION 1 ENT1 OWNS 0 : MANY ENT_B
MIGRATES E1K SET EB_BB = BB ;
ALTER ENTITY ENT_B ADD
PRIMARY KEY EBK = AA EB_BB ;
CREATE RELATION 1 ENT1 CONTAINS 0 : MANY
ENT8 ; MIGRATES E1K SET BB=BB ;
CREATE RELATION 1 ENT8 IS_PART_OF 0 : MANY
ENT_B ;

Successful Completion Criteria for Test: The execution status
will be reported as successful for each NDDL command listed in
COPENT1.OUT.

2.98 NDDL62 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL62 - Runtime

Objective: This test case will execute the NDDL command "COMBINE ENTITY" to generate NDDL commands on a specified file. These NDDL commands will physically combine two entities which exist in two separate models. All relations, keys, aliases and keywords associated with the entity are generated, except descriptions, as this option has been excluded.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 4 minutes

Special Resource Considerations : This test requires the data file CMBENT.DAT which contains the NDDL command "COMBINE ENTITY". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL CMBENT

Expected Test Results: : Each NDDL command and its execution status will be reported in the file CMBENT.OUT.

The following NDDL commands to create the new model will be generated on file CMBENT.FIL.

```
ALTER MODEL SALTY ;
CREATE ATTRIBUTE ATT_B DOMAIN NUMERIC_ID
;
ALTER ENTITY ENT3 ADD OWNED ATTRIBUTE ATT_B ;
CREATE ALIAS ENTITY ENT3 IS ENT_B ;
CREATE RELATION 1 ENT1 OWNS 0 : MANY ENT3 MIGRATES E1K
SET EB_BB = BB ;
ALTER ENTITY ENT3 ADD PRIMARY KEY EBK = EB_BB AA ;
/* THIS MAY FAIL, IF KEY NAME ALREADY EXISTS IN THE NEW ENTITY */
;
```

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in CMBENT.OUT.

2.99 NDDL63 - Runtime

Subsystem : CDM Release: 3.0
Test Name and Number : NDDL63 - Runtime

Objective: This test case will execute the NDDL command "COPY MODEL" to generate NDDL commands on a specified file. These NDDL commands will create a new model and all of its associated entities, attributes, relations, descriptions, keys and keywords that is a copy of an existing model.

Resource Requirements

Number of terminals : 1
S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test requires the data file COPMOD.DAT which contains the NDDL command "COPY MODEL". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPMOD

Expected Test Results: : Each NDDL command and its execution status will be reported in the file COPMOD.OUT.

The following NDDL commands to create the new model will be generated on file COPMOD.FIL.

```
CREATE MODEL AUGIE_COPY ;
CREATE ATTRIBUTE AA DOMAIN CHARACTER_NAME
KEYWORD AA_KEYWORD ;
CREATE ATTRIBUTE ATT_A DOMAIN CHARACTER_NAME
KEYWORD ATTA_BOY ;
DESCRIBE DEFINITION OF ATTRIBUTE ATT_A
"
THIS IS AN ALIAS OF ATT_A IN MODEL AUGIE_MOD
"
;
CREATE ALIAS ATTRIBUTE
ATT_A IS ATTA_ALIAS ;
CREATE ATTRIBUTE BB DOMAIN UNDEFINED
;
DESCRIBE DEFINITION OF ATTRIBUTE BB
```

```
"
BB IS AN ATTRIBUTE IN MODEL AUGIE_MOD
"

;
DESCRIBE EXAMPLE OF ATTRIBUTE BB
"
A RONCO CODAMATIC IS AN EXAMPLE OF A BB ATTRIBUTE
"

;
CREATE ALIAS ATTRIBUTE
    BB IS BB_ALIAS ;
CREATE ATTRIBUTE DD DOMAIN NUMERIC_ID
;
CREATE ATTRIBUTE ATT_8 DOMAIN UNDEFINED
;
CREATE ATTRIBUTE CC DOMAIN CHARACTER_NAME
    KEYWORD CC KEYWORD ;
CREATE ATTRIBUTE ATT_B DOMAIN NUMERIC_ID
;
CREATE ATTRIBUTE ATT_C DOMAIN NUMERIC_ID
;
DESCRIBE DEFINITION OF ATTRIBUTE ATT_C
"
ATT_C IS A PRIMARY ATTRIBUTE IN MODEL AUGIE_MOD
"

;
CREATE ATTRIBUTE PAY_TYPE DOMAIN CHARACTER_NAME
;
CREATE ATTRIBUTE EMP_NO DOMAIN NUBERIC_ID
;

/* The following is a list of attributes that are
not owned by any entity in the model */

CREATE ENTITY ENT_A
    ALTER ENTITY ENT_A ADD OWNED ATTRIBUTE AA ATT_A ;
CREATE ENTITY ENT_B ;
    ALTER ENTITY ENT_B ADD OWNED ATTRIBUTE ATT_B ;
CREATE ENTITY ENT_C ;
    ALTER ENTITY ENT_C ADD OWNED ATTRIBUTE CC ATT_C ;
    ALTER ENTITY ENT_C ADD KEYWORD K3 K10 ;
CREATE ENTITY ENT1 ;
    ALTER ENTITY ENT1 ADD OWNED ATTRIBUTE BB DD ;
CREATE ENTITY ENT8 ;
    ALTER ENTITY ENT8 ADD OWNED ATTRIBUTE ATT_8 ;
CREATE ENTITY ENT6 ;
    CREATE ENTITY EMPLOYEE ;
ALTER ENTITY EMPLOYEE ANDD OWNED ATTRIBUTE PAY_TYPE EMP_NO
;
CREATE ENTITY ANNUAL ;
CREATE ENTITY BIANNUAL ;
ALTER ENTITY ENT_A ADD
    PRIMARY KEY EAK = AA /* THIS MAY FAIL, IF KEY
    NAME ALREADY
    EXISTS IN THE NEW ENTITY */ ;
ALTER ENTITY ENT1 ADD
    PRIMARY KEY E1K = BB /* THIS MAY FAIL, IF KEY
    NAME ALREADY
    EXISTS IN THE ' ' ENTITY */ ;
```



```
ALTER ENTITY EMPLOYEE ADD PRIMARY KEY EMPKEY=EMP_NO
/* THIS MAY FAIL, IF KEY NAME ALREADY EXISTS IN THE
   NEW ENTITY */ ;
/* ALTER ENTITY, ADD KEYS COULD BE REDUNDANT
   IF ENTITY WAS A CATEGORY MEMBER */
CREATE RELATION 1 ENT_A HAS 0 : MANY ENT_B MIGRATES EAK
SET AA = AA
;
CREATE RELATION 1 ENT1 OWNS 0 : MANY ENT_B MIGRATES E1K
SET EB_BB = BB
;
ALTER ENTITY ENT_B ADD PRIMARY KEY EBK = AA EB_BB ;
CREATE RELATION 1 ENT1 CONTAINS 0 : MANY ENT8 MIGRATES E1K SET
EB_BB = BB
;
ALTER ENTITY ENT_B ADD PRIMARY KEY EBK = EB_BB AA ;
CREATE RELATION 1 ENT8 IS_PART_OF 0 : MANY ENT_B
MIGRATES E1K SET
BB =BB
;
CREATE RELATION 1 ENTB IS_PART_OF 0 : MANY ENT_B
;
CREATE RELATION 1 ENT_B USES 0 : MANY ENT_C MIGRATES EBK SET
EC_AA = AA EC_BB = EB_BB
KEYWORD USES KW USES KEYWORD ;
DESCRIBE DEFINITION OF RELATION ENT_B USES ENT_C
"ENT_B USES ENT_C IS A RELATION IN AUGIE_MOD"
;
DESCRIBE SOURCE OF RELATION ENT_B USES ENT_C
"SOURCE OF RELATION IS A CRAZED ANALYST"
;
ALTER ENTITY ENT_C ADD PRIMARY KEY ECK = CC EC_AA CC ;
CREATE RELATION 1 ENT_B IDENTIFIES 0 : MANY ENT6
MIGRATES EBK SET
AA = AA E6_BB = EB_BB
;
ALTER ENTITY ENT6 ADD PRIMARY KEY E6K = E6_BB ;
CREATE INCOMPLETE CATEGORY PAYROLL OF EMPLOYEE
DISCRIMINATED BY PAY_TYPE INTO CATEGORY ANNUAL IF 'A'
EMP_NO = EMP_NO CATEGORY BIANNUAL IF 'B' EMP_NO =
EMP_NO ;
ALTER ENTITY ANNUAL ADD PRIMARY KEY EMPKEY = EMP_NO ;
ALTER ENTITY BIANNUAL ADD PRIMARY KEY EMPKEY = EMP_NO ;
HALT ;
```

Successful Completion Criteria for Test: The execution status will be reported as successful for each NDDL command listed in COPMOD.OUT.

2.100 NDDL64 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL64 - Runtime

Objective: This test case will execute the NDDL command "COPY MODEL" four times, which will generate the NDDL to copy IDEF1 models with different combinations of exceptions.

This first command generates a comprehensive list of entities and relations in NDDL format as attributes have been excepted.

The second command generates a comprehensive list of attributes in NDDL format as entities have been excepted.

The third command generates all keyed attributes, as non-keyed attributes have been excepted. All entities are generated. All relations are generated without the migrates...set phase as inherited attributes are also excepted.

Finally, the fourth command excepts two entities. The model is copied without any reference to the excepted entities.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor
CDM File/Module Processing
Capabilities
NDDL command processors

Estimated Time for Test : 2 minutes

Special Resource Considerations : This test case requires the data file COPMOD1.DAT. In addition, test case NDDL54 must have executed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL COPMOD1

Expected Test Results: : The results of this runtime test will be 4 files which will contain the NDDL necessary to copy a specified model with different exceptions. NDDL generated on these files will be:

COPMOD1.FIL

```
CREATE MODEL AUGIE_MOD_1 ;
CREATE ENTITY ENT_A ;
CREATE ENTITY ENT_B ;
CREATE ENTITY ENT_C ;
CREATE ENTITY ENT1 ;
CREATE ENTITY ENT8 ;
CREATE ENTITY ENT6 ;

CREATE ENTITY EMPLOYEE ;
CREATE ENTITY ANNUAL ;
CREATE ENTITY BIANNUAL ;
/* ALTER ENTITY, ADD KEYS COULD BE REDUNDANT
   IF ENTITY WAS A CATEGORY MEMBER */
CREATE RELATION 1 ENT_A HAS 0 : MANY ENT_B ;
CREATE RELATION 1 ENT1 OWNS 0 : MANY ENT_B ;
CREATE RELATION 1 ENT1 CONTAINS 0 : MANY ENT8 ;
CREATE RELATION 1 ENT8 IS_PART_OF 0 : MANY ENT_B ;
CREATE RELATION 1 ENT_B USES 0 : MANY ENT_C ;
CREATE RELATION 1 ENT_B IDENTIFIES 0 : MANY ENT6 ;
CREATE INCOMPLETE CATEGORY PAYROLL OF EMPLOYEE
DISCRIMINATED BY PAY_TYPE INTO CATEGORY ANNUAL IF
'A'EMP_NO =
EMP_NO CATEGORY BIANNUAL IF 'B' EMP_NO = EMP_NO ;
HALT ;
```

```
COPMOD2.FIL
CREATE MODEL AUGIE_MOD_2 ;
CREATE ATTRIBUTE AA DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE ATT_A DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE BB DOMAIN UNDEFINED ;
CREATE ATTRIBUTE DD DOMAIN NUMERIC_ID ;
CREATE ATTRIBUTE ATT_8 DOMAIN UNDEFINED ;
CREATE ATTRIBUTE CC DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE ATT_B DOMAIN NUMERIC_ID ;
CREATE ATTRIBUTE ATT_C DOMAIN NUMERIC_ID ;
CREATE ATTRIBUTE PAY_TYPE DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE EMP_NO DOMAIN NUMERIC_ID ;
/* The following is a list of attributes that
   are not owned by any entity in the model */

HALT ;
```

```
COPMOD3.FIL
CREATE MODEL AUGIE_MOD_3 ;
CREATE ATTRIBUTE AA DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE BB DOMAIN UNDEFINED ;
CREATE ATTRIBUTE CC DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE EMP_NO DOMAIN NUMERIC_ID ;

/* The following is a list of attributes that
   are not owned by any entity in the model */

CREATE ENTITY ENT_A ;
ALTER ENTITY ENT_A ADD OWNED ATTRIBUTE AA ;
CREATE ENTITY ENT_B ;
CREATE ENTITY ENT_C ;
ALTER ENTITY ENT_C ADD OWNED ATTRIBUTE CC ;
CREATE ENTITY ENT1 ;
```

```
ALTER ENTITY ENT1 ADD OWNED ATTRIBUTE BB ;
CREATE ENTITY ENT8 ;
CREATE ENTITY ENT6 ;

CREATE ENTITY EMPLOYEE ;
ALTER ENTITY EMPLOYEE ADD OWNED ATTRIBUTE EMP_NO ;
CREATE ENTITY ANNUAL ;
CREATE ENTITY BIANNUAL ;
/* ALTER ENTITY, ADD KEYS COULD BE REDUNDANT
   IF ENTITY WAS A CATEGORY MEMBER */ CREATE RELATION 1
ENT_A HAS 0 :
    MANY ENT_B ;
CREATE RELATION 1 ENT1 OWNS 0 : MANY ENT_B ;
CREATE RELATION 1 ENT1 CONTAINS 0 : MANY ENT8 ;
CREATE RELATION 1 ENT8 IS PART OF 0 : MANY ENT_B ;
CREATE RELATION 1 ENT_B USES 0 : MANY ENT_C ;
CREATE RELATION 1 ENT_B IDENTIFIES 0 : MANY ENT6 ;
CREATE INCOMPLETE CATEGORY PAYROLL OF EMPLOYEE
    DISCRIMINATED BY PAY_TYPE INTO CATEGORY ANNUAL IF 'A'
EMP_NO =
    EMP_NO CATEGORY BIANNUAL IF 'B' EMP_NO =EMP_NO ;
HALT ;

COPMOD4.FIL
CREATE MODEL AUGIE_MOD 4 ;
CREATE ATTRIBUTE AA DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE ATT_A DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE BB DOMAIN UNDEFINED ;
CREATE ATTRIBUTE DD DOMAIN NUMERIC ID ;
CREATE ATTRIBUTE ATT_8 DOMAIN UNDEFINED ;
CREATE ATTRIBUTE PAY_TYPE DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE EMP_NO DOMAIN NUMERIC_ID ;

/* The following is a list of attributes that are not owned
   by any entity in the model */

CREATE ENTITY ENT_A ;
ALTER ENTITY ENT_A ADD OWNED ATTRIBUTE AA ATT_A ;
CREATE ENTITY ENT1 ;
ALTER ENTITY ENT1 ADD OWNED ATTRIBUTE BB DD ;
CREATE ENTITY ENT8 ;
ALTER ENTITY ENT8 ADD OWNED ATTRIBUTE ATT_8 ;
CREATE ENTITY EMPLOYEE ADD OWNED ATTRIBUTE PAY_TYPE EMP_NO ;
CREATE ENTITY ANNUAL ;
CREATE ENTITY BIANNUAL ;
ALTER ENTITY ENT_A ADD
    PRIMARY KEY EAK = AA /* THIS MAY FAIL, IF KEY NAME ALREADY
    EXISTS IN THE NEW ENTITY */ ;
ALTER ENTITY ENT1 ADD
    PRIMARY KEY E1K = BB /*THIS MAY FAIL, IF KEY NAME
    ALREADY EXISTS
    IN THE NEW ENTITY */ ;
ALTER ENTITY EMPLOYEE ADD
    PRIMARY KEY EMPKEY = EMP_NO
    /* THIS MAY FAIL, IF KEY_NAME ALREADY EXISTS IN THE NEW
    ENTITY
    */ ;
/* ALTER ENTITY, ADD KEYS COULD BE REDUNDANT
   IF ENTITY WAS A CATEGORY MEMBER */ CREATE RELATION 1
```

```
ENT1 CONTAINS 0
      : MANY ENT8 MIGRATED E1K SET
      BB =BB ;
CREATE INCOMPLETE CATEGORY PAYROLL OF EMPLOYEE
      DISCRIMINATED BY PAY_TYPE INTO CATEGORY ANNUAL IF 'A'
EMP_NO =
      EMP NO CATEGORY BIANNUAL IF 'B' EMP_NO =EMP_NO ;
ALTER ENTITY ANNUAL ADD
      PRIMARY EKY EMPKEY = EMP_NO ;
ALTER ENTITY BIANNUAL ADD
      PRIMARY KEY EMPKEY = EMP_NO ;
HALT ;
```

Successful Completion

Criteria for Test: : The execution status will be reported
as successful for each NDDL command listed in COPMOD1.OUT.

2.101 NDDL65 - Runtime

Subsystem : CDM Release: 3.0

Test Name and Number : NDDL65 - Runtime

Objective: This test case will execute the NDDL command "MERGE MODEL" to generate NDDL commands on a specified file. These NDDL commands will physically merge two IDEF1 models, either combining or copying entities, along with associated attributes, keys, and relations. Keywords, aliases and descriptions will not be generated as they have been excepted.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing

Capabilities
NDDL command processors

Estimated Time for Test : 5 minutes

Special Resource Considerations : This test requires the data file MRGMOD.DAT which contains the NDDL command "MERGE MODEL". In addition, test case NDDL54 must have completed successfully.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL MRGMOD

Expected Test Results: Each NDDL comand and its execution status will be reported in the file named above as MRGMOD.OUT.

The following NDDL commands to merge the models are generated on file MRGMOD.FIL.

```
ALTER MODEL AUGIE MOD ;
CREATE ATTRIBUTE EE DOMAIN CHARACTER_NAME ;
CREATE ATTRIBUTE XX DOMAIN CHARACTER_NAME ;
ALTER ENTITY ENT1 ADD OWNED ATTRIBUTE EE XX ;
ALTER ENTITY ENT1 ADD
    PRIMARY KEY E1K = EE XX /* THIS MAY FAIL,
IF KEY NAME
    ALREADY EXISTS IN THE NEW ENTITY */ ;
CREATE ENTITY ENT3 OWNED ATTRIBUTE CC
    /* THIS ATTRIBUTE MAY BE OWNED IN TARGET
MODEL */ ;
CREATE RELATION 1 ENT1 R1 0 : MANY ENT3
```

```

        MIGRATES E1K SET
        EE = EE XX = XX ;
    ALTER ENTITY ENT3 ADD
        PRIMARY KEY E3K3 = EE CC ;

    CREATE ENTITY ENT4 OWNED ATTRIBUTE DD
        /* THIS ATTRIBUTE MAY BE OWNED IN TARGET
MODEL */ ;
    CREATE RELATION 1 ENT3 R3 0 : MANY ENT4
        MIGRATES E3K3 SET
        EE = EE CC = CC ;
    ALTER ENTITY ENT4 ADD
        PRIMARY KEY E4K4 = CC DD ;

    CREATE ATTRIBUTE YY DOMAIN CHARACTER NAME ;
    CREATE ATTRIBUTE ZZ DOMAIN CHARACTER NAME ;
    ALTER ENTITY ENT_C ADD OWNED ATTRIBUTE YY ZZ ;
    CREATE RELATION 1 ENT3 USES 0 : MANY ENT_C
        MIGRATES E3K3 SET
        EE = EE CC = CC ;
    ALTER ENTITY ENT_C ADD
        PRIMARY KEY ECKC = EE YY ;

    CREATE ENTITY ENT_F ;
    CREATE INCOMPLETE CATEGORY CAT1 OF ENT_C DISCRIMINATED
BY ZZ
        INTO CATEGORY ENT_F IF 'B' YY = YY EE = EE ;
    ALTER ENTITY ENT_F ADD
        PRIMARY KEY ECKC = YY EE ;
```

Successful Completion Criteria for Test:

The execution status will be reported as successful for each NDDL command listed in MRGMOD.OUT.

2.102 NDDL66 - Runtime

Subsystem: CDM

Release: 3.0

Test Name and Number: NDDL66 - Runtime

Objective:

This test case will drop the IDEF1 conceptual schema models that were created for testing NDDL modeling commands. This should be the concluding test in the series NDDL54 through NDDL65.

Resource Requirements

Number of terminals : 1

S/W Requirements : ORACLE CDM DATABASE
CDMP: Distributed Request Supervisor

CDM File/Module Processing Capabilities
NDDL command processors

Estimated Time for Test : 15 minutes

Special Resource Considerations : This test requires the data file DROPMOD.DAT which contains the NDDL commands. The test will also require access to the IISS CDM.

Test Definition

Method of Performing Test : With the ORACLE CDM database available, at the VAX/VMS prompt (\$) type the test inputs defined below.

Test Inputs : \$ @NDDL DROPMOD

Expected Test Results: : Each NDDL command and its execution status will be reported in the file DROPMOD.OUT. All seven models will be dropped from the CDM. This includes all attributes, entities, relations, keys, keywords, descriptions, etc. associated with the model.

Successful Completion Criteria for Test:

The execution status will be reported as successful for each NDDL command listed in DROPMOD.OUT.